

Hybrid Gate-Based and Annealing Quantum Computing for Large-Size Ising Problems

Chen-Yu Liu

d10245003@g.ntu.edu.tw

National Taiwan University

Hsi-Sheng Goan

National Taiwan University

Research Article

Keywords: Quantum optimization, Quantum annealing, Variational quantum eigensolver, Portfolio optimization

Posted Date: July 31st, 2024

DOI: <https://doi.org/10.21203/rs.3.rs-4707131/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Hybrid Gate-Based and Annealing Quantum Computing for Large-Size Ising Problems

Chen-Yu Liu^{1,*} and Hsi-Sheng Goan^{1,2,3,4,†}

¹*Graduate Institute of Applied Physics, National Taiwan University, Taipei 106319, Taiwan*

²*Department of Physics and Center for Theoretical Physics,
National Taiwan University, Taipei 106319, Taiwan*

³*Center for Quantum Science and Engineering, National Taiwan University, Taipei 106319, Taiwan*

⁴*Physics Division, National Center for Theoretical Sciences, Taipei, 106319, Taiwan*

One of the major problems of quantum computing applications is that the required number of qubits to solve a practical problem is much larger than that of today's quantum hardware. In this work, we introduce the large-system sampling approximation (LSSA) algorithm to solve Ising problems with sizes up to $N_{\text{gb}}2^{N_{\text{sb}}}$ by an N_{gb} -qubit gate-based quantum computer, and problems with sizes up to $N_{\text{an}}2^{N_{\text{sb}}}$ by a hybrid computational architecture of an N_{an} -qubit quantum annealer and an N_{gb} -qubit gate-based quantum computer. By dividing the full-system problem into smaller subsystem problems, the LSSA algorithm then solves the subsystem problems by either gate-based quantum computers or quantum annealers, and optimizes the amplitude contributions of the solutions of the different subsystems with the full-problem Hamiltonian by the variational quantum eigensolver (VQE) on a gate-based quantum computer to determine the approximated ground-state configuration. LSSA has polynomial time complexity and can be further extended to a deeper level of approximation with computational overhead that grows linearly with the problem size. The effects of different subsystem sizes, numbers of subsystems, and full problem sizes on the performance of LSSA are investigated on both simulators and real hardware. The completely new computational concept of the hybrid gate-based and annealing quantum computing architecture opens a promising possibility to investigate large-size Ising problems and combinatorial optimization problems, making practical applications by quantum computing possible in the near future.

Keywords: Quantum optimization, Quantum annealing, Variational quantum eigensolver, Portfolio optimization

I. INTRODUCTION

Accessible cloud quantum computing resources provided by several companies [1–3] makes the development of quantum computing applications practical and popular in recent years. Appropriate mapping from the original problem to quantum solvable form is required to solve problems on quantum computers. A popular choice for solving a large class of combinatorial optimization problems that can be converted into quadratic unconstrained binary optimization (QUBO) problems is mapping the problem to an Ising Hamiltonian and solving for the corresponding ground state of the Hamiltonian, which is related to the solution of the original problem [4, 5]. Applications of quantum optimization to real-world problems have been demonstrated for portfolio optimizations [6–9], industrial optimization problems [10, 11], and traveling salesman problems [12, 13]. To find the optimal solution of the corresponding Hamiltonian, the gate-based quantum computing with variational quantum eigensolver (VQE) uses parameterized gates to construct a trial state and optimizes the best set of parameters that can approximate the ground state of

the Ising Hamiltonian [14–16]. Another commonly used gate-based variational quantum algorithm (VQA) is the quantum approximate optimization algorithm (QAOA), where the trotterization of the adiabatic evolution makes it a scalable approach [17–29]. However, QAOA and VQE are designed to solve the Ising Hamiltonian problems with N_p variables on an N_p -qubit gate-based quantum computer. Thus, even the gate-based quantum computer `ibm_torino` provided recently by IBM can only solve the problem with 133 variables if we use the original VQE and QAOA algorithms.

To face this issue, several qubit-efficient methods are proposed to tackle the problem with size larger than the hardware size. Divide-and-Conquer QAOA (DC-QAOA) that was proposed for graph maximum cut (MaxCut) problems divides a large graph recursively to the size of the available hardware with each step splitting a resultant graph into exactly two subgraphs [30]. Then a classical method of the combination policy of quantum state reconstruction is used to reconstruct the solution by inputting the sampling distribution of each subgraph solution. However, DC-QAOA only applies to input graphs with connectivity less than maximum allowed qubit size, which makes it probably inapplicable to the fully-connected graph problems [30]. Additionally, QAOA-in-QAOA, also divides and conquers the graph MaxCut problem by utilizing the structure of graphs and the $\mathbb{Z}_2$ symmetry, and after the subgraphs are solved, the merging of local solutions of all subgraphs is reformulated into a new Maxcut problem. Although maximally 9-degree graphs are tested in the paper, the applicability and the performance on problems other than MaxCut,

* d10245003@g.ntu.edu.tw

† goan@phys.ntu.edu.tw

e.g., on the complete (fully-connected) graphs, are not discussed [31]. Another method of cluster-VQE shows an example in quantum chemistry that splitting qubits into different clusters and minimizing the correlations between clusters, with the help of a method called “Hamiltonian dressing”. But depends on the problem Hamiltonian (strongly correlated systems, for example), computational scaling of the algorithm can be exponential with respect to the dressing steps [32]. Besides, an encoding scheme is also proposed to find the solution by smaller hardware [33]. In both the works of QAOA-in-QAOA [31] and cluster-VQE [32], they demonstrated the possibility to integrate the community detection algorithm for selecting subgraphs or sub-Hamiltonians, thereby enhancing the performance compared to the random grouping method.

These approaches focus on using gate-based quantum computers, while a quantum annealer which usually has a larger qubit number is also a candidate to solve the large-size Ising problem [34–36]. As an example of that, D-Wave also provides a quantum-classical hybrid solver to break the full problem into small pieces of subsystem problems and then update the full problem solution locally and iteratively to obtain an approximate solution for the full problem [37–39]. Another similar research proposes a quantum local search algorithm that can solve the subsystem problems by either gate-based chips or annealing chips [40]. Although these approaches allow smaller quantum computing chips to solve the larger problem, the iterative design in the algorithms makes them only capable of solving one subsystem problem at each iteration, and introduces a causality on subsystem solutions between different iterations, and thus the subsystems cannot be solved in parallel. In most cases, the sizes of subsystem problems solved by gate-based chips are much smaller than those by annealing chips, while gate-based chips can utilize gate operations to encode information that may be helpful in a problem-solving scheme.

In this paper, we propose an approach called the large-system sampling approximation (LSSA) that tries to solve a larger-size problem via randomly sampling it into many subsystem problems solved by smaller-size quantum hardware, and then employ VQE to approximate the solution of the larger-size problem by optimizing the contributions (coefficients) of the subsystem solutions using the full-size Hamiltonian. By combining the advantages of gate-based and annealing quantum computing approaches mentioned above, our proposed LSSA comes with a hybrid structure that can solve the subsystem problems by either gate-based chips or annealing chips, and the coefficient optimization of recombining the solutions of the subsystem problems can be performed by VQE on gate-based chips. We will show that our proposed algorithm can approximately solve, in our selected parameter setting and problem testing, fully-connected random Ising problems that are $O(10^0)$ and portfolio optimization problems that are $O(10^1)$ larger in size than

the available quantum annealers and gate-based quantum computers, with pretty good performance of the results from both simulators and real hardware. With its polynomial time scaling behavior and the ability to solve subsystems in parallel, this algorithm presents a viable approach for tackling large-scale real-world problems with current quantum hardware. It’s important to note that although the proposed LSSA approach exhibit polynomial computational time scaling, akin to nearly all heuristic methods, this does not guarantee the obtained results are optimal. However, the LSSA algorithm is capable of finding solutions that are quite satisfactory for some Ising and combinatorial optimization problems.

Compared to DC-QAOA [30], which can only solve the MaxCut problem for graphs with limited connectivity, and QAOA-in-QAOA [31], which also solves the MaxCut problems but has unknown performance on fully-connected graphs, our approach can deal with the general Ising problems that could correspond to fully-connected graphs with bias terms. Compared to that of the cluster-VQE, which solves the quantum chemistry problems via smaller-size quantum hardware with a potential exponential computational overhead in dressing steps [32], the LSSA approach that solves the larger-size problem via smaller-size quantum hardware has a computational overhead growing only linearly with problem size. Compared to the D-Wave’s approach and other quantum local search approaches to solving large Ising problems, our LSSA approach can solve the subsystem problems in parallel and thus can achieve an optional speedup by parallelism.

This paper is organized as follows. we will first introduce the idea of the proposed algorithm in Sec. II. In Sec. III, we then use the method of exact diagonalization on classical computers and quantum simulators to systematically investigate how the performance ratio changes for different parameters of the proposed algorithm. After investigating the problem on quantum simulators, we will show further result obtained from real hardware in Sec. IV. In Sec. V, the potential computational advantage of LSSA will be introduced. We will discuss the results in Sec. VI, along with the proper use cases for different kinds of problems. Then we will summarize the paper and future work based on the results in Sec. VII.

II. LARGE-SYSTEM SAMPLING APPROXIMATION

Several classes of QUBO problems can be mapped into Ising problems by the concept that the binary variable $x_i \in \{0, 1\}$ can be related to the up and down spin state in a spin- $\frac{1}{2}$ system by $z_i = 1 - 2x_i$, where $z_i \in \{-1, 1\}$ is the spin variable. Variable x_i could represent the binary decision, the label of the node in a graph, or even the binary bit in a factorization problem [4, 5]. Making use of the superposition nature of the quantum system, the

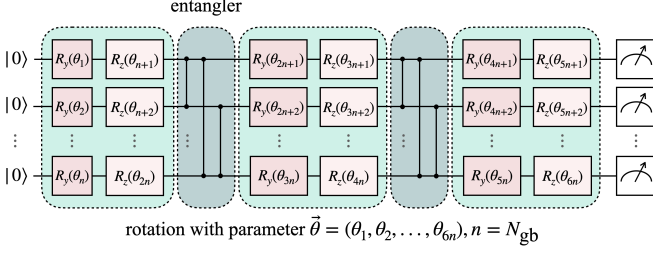


FIG. 1. Quantum circuit structure of VQE for the subsystem coefficient optimization. The initial state $|0\rangle^{\otimes N_{gb}}$ is followed by a circuit of alternating rotation layer and entanglement layer. The rotation layer is composed of parameterized single-qubit rotation gates and the entanglement layer is composed of two-qubit entangling gates. Here, in the case of a N_{gb} -qubit quantum circuit with two entanglement layers sandwiched by three rotation layers, a total of $6N_{gb}$ tunable parameters of the rotation angles of the R_y and R_z rotation gates is denoted as $\vec{\theta} = (\theta_1, \theta_2, \dots, \theta_{6N_{gb}})$.

spin system can then be solved by quantum computers. In this work, we consider Ising problems in the form:

$$H = \sum_{i,j=1}^{N_p} J_{ij} z_i z_j + \sum_{i=1}^{N_p} h_i z_i, \quad (1)$$

where z_i are the spin variables, N_p is the number of spin variables (or the problem size), and h_i and J_{ij} correspond respectively to the bias and coupling strength (weight) of the spin system with indices $i, j \in \{1, 2, \dots, N_p\}$. Here the values of h_i and J_{ij} are considered to be in the range of $(-1, 1)$.

Instead of solving the problem of the full N_p -spin system directly, we try to group the spins into smaller subsystems and then solve the problems of the subsystems first. After collecting the results of the subsystems, an estimation of the full system result will be made based on the statistical behavior of the subsystem results. Since this method is intended to approximate the solution of a large-scale full-system problem by sampling various subsystems with solvable problem size and then optimizing the amplitude contributions of the the subsystem solutions through the full-system Hamiltonian, we called it LSSA. For an N_p -spin Ising problem with the Hamiltonian of Eq. (1), consider a sampling method (described later) that picks a subsystem of N_g sites with $N_g \leq N_p$, and then the subsystem Hamiltonian can be written as:

$$H_{\text{sub}} = \sum_{i',j'=1}^{N_g} J_{i'j'} z_{i'} z_{j'} + \sum_{i'=1}^{N_g} h_{i'} z_{i'}, \quad (2)$$

where the subsystem sites have been labeled with indices i' and j' , where $i', j' \in \{1', 2', \dots, N_g'\}$. After we solved the relatively simple (compare to H) eigenvalue problem of H_{sub} , the corresponding ground state $|\text{GS}_{\text{sub}}'\rangle = [s_{1'}, s_{2'}, \dots, s_{N_g'}]^T \in \{-1, 1\}^{N_g}$ is obtained,

where the state with the $'$ symbol indicates that the indices of the associated spin variables are the subsystem labels and on each site $s_{i'} \in \{-1, 1\}$. Next, we relabel the subsystem labels i', j' back to the full-system labels i, j , and thus obtain a part of the spin configuration of the full system, that is, a vector of length N_p with N_g non-empty elements. For example, the subsystem ground state obtained from the n th sampled subsystem Hamiltonian could be written as $|\text{GS}_{\text{sub}}^{(n)}\rangle = [s_1^{(n)}, 0, s_3^{(n)}, s_4^{(n)}, 0, \dots, s_{N_p}^{(n)}]^T \in \{-1, 0, 1\}^{N_p}$, where the $(N_p - N_g)$ sites (elements) with value 0 representing the empty information correspond to the sites that are not picked for the subsystem. The random subsystem sampling will be performed N_s times in the following manner. The subsystems of size- N_g will be picked randomly and non-repetitively until all of the variables (sites) in a full problem are picked at least once. After that, if the number of the subsystems is below N_s , the next round of non-repetitive random sampling will be made until all of the variables are picked at least twice or until the number of subsystems is N_s . The sampling procedure will be carried out in this fashion until all N_s subsystems are formed, and thus all the variables will be picked at least $\lfloor (N_s \times N_g) / N_p \rfloor$ times. The condition $N_s \times N_g \geq N_p$ must be satisfied to cover all the variables in a problem at least once.

After obtaining the results from the N_s subsystems, we construct a vector denoted as $|\mathcal{S}^{\text{wc}}\rangle \in \mathbb{R}^{N_p}$, which represents the weighted ground states of these subsystems:

$$|\mathcal{S}^{\text{wc}}\rangle = \sum_{n=1}^{N_s} C^{(n)} |\text{GS}_{\text{sub}}^{(n)}\rangle. \quad (3)$$

This vector is generated by summing the ground state result of each sampling, $|\text{GS}_{\text{sub}}^{(n)}\rangle$, weighted by an undetermined coefficient $C^{(n)} \in (0, 1)$, signifying the contribution of the corresponding n th subsystem. The m th element $|\mathcal{S}^{\text{wc}}\rangle_m$ of the constructed vector $|\mathcal{S}^{\text{wc}}\rangle$ can be expressed as $C^{(1)} s_m^{(1)} + C^{(2)} s_m^{(2)} + \dots + C^{(n)} s_m^{(n)} + \dots + C^{(N_s)} s_m^{(N_s)}$ representing the weighted solution configuration of the m th site. The value of $C^{(n)}$ will be determined by an optimization procedure that takes into account the amplitude contributions of the solution of the corresponding subsystem n to the ground-state solution of the full-size system. This procedure will be described later. Since the value of the m th element $|\mathcal{S}^{\text{wc}}\rangle_m$ should also be either +1 or -1, we take the sign of $|\mathcal{S}^{\text{wc}}\rangle_m$ to indicate the accumulated behavior of the m th site over the N_s subsystems. This could then resolve the value of $|\mathcal{S}^{\text{wc}}\rangle_m$ to be either +1 or -1, and could also potentially offer insight into the behavior of the complete system.

Following this line of thought, we approximate the ground-state configuration of the full-size system Hamiltonian to be the state resulting from taking the sign of each element in the $|\mathcal{S}^{\text{wc}}\rangle$ vector:

$$|\text{GS}_{\text{full}}\rangle \approx |\text{sign}(\mathcal{S}^{\text{wc}})\rangle, \quad (4)$$

where $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle \in \{-1, 1\}^{N_p}$. The m th element $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle_m$ of the vector $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle$ is +1 if $|\mathcal{S}^{\text{wc}}\rangle_m \geq 0$, and -1 if $|\mathcal{S}^{\text{wc}}\rangle_m < 0$. Note that the assignment of $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle_m = +1$ for $|\mathcal{S}^{\text{wc}}\rangle_m = 0$ is for the case of frustration. As the subsystem size approaches that of the full system, i.e., as $N_g \rightarrow N_p$, it is anticipated that $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle$ will converge towards $|\text{GS}_{\text{full}}\rangle$.

To determine the optimal contributions or coefficients $C^{(n)}$ from each subsystem, which bring $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle$ closer to the true ground state, we employ the VQE algorithm to minimize the ground state energy of the entire system governed by the full-system Hamiltonian H . We leverage a mapping of $C^{(n)}$ into a quantum state by a parameterized quantum circuit by associating N_s coefficients with the measured probability $|\langle \phi_n | \psi(\vec{\theta}) \rangle|^2$, where $|\psi(\vec{\theta})\rangle$ is the parameterized quantum state and $|\phi_n\rangle$ is a computational basis state. Mathematically, this can be expressed as:

$$C^{(n)}(\vec{\theta}) = |\langle \phi_n | \psi(\vec{\theta}) \rangle|^2, \quad (5)$$

where $\vec{\theta}$ encapsulates a set of tunable parameters in the parameterized quantum circuit applied in the VQE algorithm. Thus a gate-based quantum system with $N_{\text{gb}} = \lceil \log_2 N_s \rceil$ qubits is required. Notably, since $n \in \{1, 2, \dots, N_s\}$ and $k \in \{1, 2, \dots, 2^{N_{\text{gb}}}\}$ for the basis state $|\phi_k\rangle$, the choice of $N_{\text{gb}} = \lceil \log_2 N_s \rceil$ ensures that $2^{N_{\text{gb}}-1} < N_s \leq 2^{N_{\text{gb}}}$.

The objective or cost function of the VQE algorithm is formulated as the energy corresponding to the estimated N_p -qubit ground state $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle$:

$$\text{Cost}(\mathbf{C}(\vec{\theta})) = \langle \text{sign}(\mathcal{S}^{\text{wc}}(\vec{\theta})) | H | \text{sign}(\mathcal{S}^{\text{wc}}(\vec{\theta})) \rangle, \quad (6)$$

where $\mathbf{C}(\vec{\theta}) = (C^{(1)}(\vec{\theta}), C^{(2)}(\vec{\theta}), \dots, C^{(N_s)}(\vec{\theta}))$, and H is the size- N_p Hamiltonian of Eq. (1). It is important to note that the $\vec{\theta}$ dependence of $|\text{sign}(\mathcal{S}^{\text{wc}}(\vec{\theta}))\rangle$ arises from the mapping of the coefficients $C^{(n)}(\vec{\theta})$ into the parameterized quantum state through Eq. (5). In contrast to the conventional VQE approach where the parameterized quantum state from the quantum circuit is directly used to evaluate the expectation value of the Hamiltonian, our formulation differs. In our approach, the N_{gb} -qubit VQE quantum circuit is employed to generate the N_s subsystem coefficients through the parameterized quantum state $|\psi(\vec{\theta})\rangle$ by Eq. (5), and then the expectation value of the Hamiltonian of the cost function Eq. (6) is computed using the estimated N_p -qubit ground state $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle$ obtained via Eq. (3). We will give an example of how to obtain $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle$ explicitly later.

The quantum circuit used in our VQE is chosen to be a heuristic hardware-efficient circuit, and the circuit structure consists of alternating rotation layers and entanglement layers. The rotation layer is composed of single-qubit rotation gates and the entanglement layer is composed of two-qubit entangling gates. A typical quantum circuit shown in Fig. 1 contains two entanglement layers sandwiched by three rotation layers. The

tunable parameters in this N_{gb} -qubit quantum circuit are the rotation angles of the R_y and R_z rotation gates in the rotational layers, and the circuit has a total of $6N_{\text{gb}}$ tunable parameters of the rotation angles, that is, $\vec{\theta} = (\theta_1, \theta_2, \dots, \theta_{6N_{\text{gb}}})$. The variational search of $C^{(n)}$ by minimizing the cost function, Eq. (6), on hybrid quantum and classical computers makes the task of finding N_s (which is comparable to $2^{N_{\text{gb}}}$) $C^{(n)}$ coefficients to the task of finding $6N_{\text{gb}}$ parameters (gate angles). Compared to the optimization of the same problem by classical computer alone that will normally require N_s ($\approx 2^{N_{\text{gb}}}$) parameters, the VQE approach has an exponential advantage in parameter saving which is particularly appealing when N_s is large.

Additionally, it is also possible to have a different circuit structure of VQE, with different numbers of layers, configurations of gate connectivity, and types of rotation and entangling gates. These can all be considered as hyperparameters that can be varied in VQE. Appendix B shows the effect of these hyperparameters on our results.

In this formulation, a single N_{gb} -qubit gate-based quantum computer can deal with QUBO problems up to $N_{\text{gb}}2^{N_{\text{gb}}}$ variables as each subsystem size problem can have up to N_{gb} variables to be tackled by algorithms like QAOA, and maximally $2^{N_{\text{gb}}}$ subsystems can be encoded into VQE for the coefficient contribution optimization. Interestingly, it is possible to use a quantum annealer to deal with a significantly larger size of subsystem problems and then encode them to a gate-based quantum computer for the optimization of the coefficient contributions of the subsystems. In this case, a QUBO problem with size up to $N_{\text{an}}2^{N_{\text{gb}}}$ variables could be handled, where N_{an} is the size of the subsystems that can be dealt with by the quantum annealer. The corresponding results will be discussed later. The flow of LSSA including the coefficient contribution optimization is shown in Fig. 2. In this paper, $\vec{\theta}$ are updated by the COBYLA optimization method up to 200 iterations.

As a simple example to illustrate how to obtain $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle$ in LSSA, let us consider a problem of size $N_p = 4$, subsystem size $N_g = 2$, and the number of subsystems $N_s = 4$. Suppose the results of the ($N_s = 4$) solved subsystem ground states with length $N_p = 4$ are

$$|\text{GS}_{\text{sub}}^{(1)}\rangle = [1, -1, 0, 0]^T, \quad (7)$$

$$|\text{GS}_{\text{sub}}^{(2)}\rangle = [0, 0, 1, -1]^T, \quad (8)$$

$$|\text{GS}_{\text{sub}}^{(3)}\rangle = [-1, 0, 0, 1]^T, \quad (9)$$

$$|\text{GS}_{\text{sub}}^{(4)}\rangle = [0, 1, 1, 0]^T. \quad (10)$$

Note that there are $(N_p - N_g) = 2$ elements with value 0 in each ground-state of Eqs. (7)-(10) corresponding to the empty information of the site that is not selected in the sampled subsystem configuration. In this case, $|\mathcal{S}^{\text{wc}}\rangle$

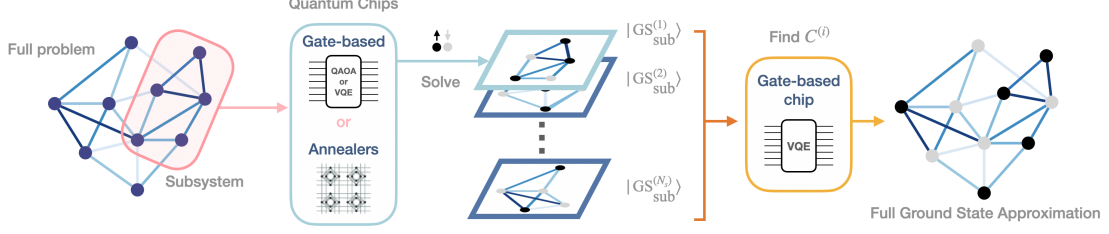


FIG. 2. Schematic flow chart of LSSA. After a procedure of random grouping of the spin variables from the full-system problem to form subsystems, the subsystem ground-state problems are solved by gate-based or annealing quantum chips. The resulting subsystem ground states are used to construct a trial state vector $|\mathcal{S}^{\text{wc}}\rangle$ with undetermined coefficients (amplitudes) described in Eq. (3), and then VQE is used to find the optimized coefficients with the cost function of Eq. (6).

from Eq. (3) becomes

$$\begin{aligned}
 |\mathcal{S}^{\text{wc}}\rangle &= \sum_{n=1}^{N_s=4} C^{(n)} |\text{GS}_{\text{sub}}^{(n)}\rangle \\
 &= [C^{(1)} s_1^{(1)} + C^{(2)} s_1^{(2)} + C^{(3)} s_1^{(3)} + C^{(4)} s_1^{(4)}, \\
 &\quad C^{(1)} s_2^{(1)} + C^{(2)} s_2^{(2)} + C^{(3)} s_2^{(3)} + C^{(4)} s_2^{(4)}, \\
 &\quad C^{(1)} s_3^{(1)} + C^{(2)} s_3^{(2)} + C^{(3)} s_3^{(3)} + C^{(4)} s_3^{(4)}, \\
 &\quad C^{(1)} s_4^{(1)} + C^{(2)} s_4^{(2)} + C^{(3)} s_4^{(3)} + C^{(4)} s_4^{(4)}]^T \\
 &= [C^{(1)} - C^{(3)}, -C^{(1)} + C^{(4)}, \\
 &\quad C^{(2)} + C^{(4)}, -C^{(2)} + C^{(3)}]^T. \tag{11}
 \end{aligned}$$

where we have used the values of $s_m^{(n)}$ in each ground-state of $|\text{GS}_{\text{sub}}^{(n)}\rangle$ in Eqs. (7)-(10) to obtain Eq. (11). After the VQE optimization procedure to determine the values of the coefficient $C^{(n)}$ described earlier in Eqs. (5) and (6), suppose we have $C^{(1)} - C^{(3)} = 0$, $-C^{(1)} + C^{(4)} < 0$, $C^{(2)} + C^{(4)} > 0$, and $-C^{(2)} + C^{(3)} < 0$. Then from Eq. (4), the approximated ground state reads $|\text{GS}_{\text{full}}\rangle \approx |\text{sign}(\mathcal{S}^{\text{wc}})\rangle = [1, -1, 1, -1]^T$.

In summary, the conventional QAOA or VQE approach where the parameterized quantum state from the quantum circuit is directly used to evaluate and optimize the expectation value of the Ising problem Hamiltonian, requires a qubit usage corresponding to the number of variables in the problem Hamiltonian and thus is not suitable for the final stage optimization as the full-size problem Hamiltonian is employed. In contrast, our VQE formulation differs fundamentally and the qubit usage of our formulation is only $N_{\text{gb}} = \lceil \log_2 N_s \rceil$. The purpose of that last VQE phase in our LSSA method shown in Fig. 2 is to optimize the coefficient contribution of each subsystem solution to the full-system ground state solution [see Eqs. (3), (4) and (5)] using a cost function with the full-system Hamiltonian, Eq. (6). Specifically in our approach, the N_{gb} -qubit VQE quantum circuit is employed to generate the N_s subsystem coefficients through the parameterized quantum state $|\psi(\vec{\theta})\rangle$ by Eq. (5), and then the expectation value of the Hamiltonian H for the cost function $\text{Cost}(\mathbf{C}(\vec{\theta}))$ is computed

using the estimated size- N_p ground state $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle$ through Eq. (6). Note that (i) the approximate size- N_p ground state $|\text{sign}(\mathcal{S}^{\text{wc}})\rangle$ is constructed by summing the subsystem ground state solutions obtained using only N_g qubits through the process we have described above, (ii) the qubit requirement for the VQE is $N_{\text{gb}} = \lceil \log_2 N_s \rceil$, and since $N_s \sim O(N_p/N_g)$, one can conclude that N_{gb} is a much smaller number than N_p . The number of qubits used in the VQE phase of our LSSA approach is in contrast to that of using QAOA or VQE to directly solve for the ground state of a size- N_p Ising problem, which would normally require N_p qubits. Therefore, the use of VQE in the last phase of the LSSA method is regarded as a strength in our approach to tackle a large-size Ising problem by using small-size quantum hardware.

III. EXACT DIAGONALIZATION AND SIMULATORS

A. Random Ising Problems

In this paper, all the random Ising problems considered are fully-connected unless we specifically mention. First, we test the LSSA algorithm for the random Ising problems in the form of Eq. (1) with problem sizes from $N_p = 8$ to $N_p = 20$. For each problem size, 100 random Ising Hamiltonians are constructed and each is solved by sampling subsystems $\lceil \frac{2N_p}{N_g} \rceil$ times with subsystem size $N_g = 3N_p/4$, $N_g = N_p/2$ and $N_g = N_p/4$, respectively. The ranges of the random couplings J_{ij} and the biases h_i are both constrained into the range $(-1, 1)$. We use the performance ratio R_p , defined as the ration of the ground state energy (GSE) obtained by LSSA to GSE obtained by the best affordable method, as an indicator for performance. The higher of the value of R_p approaches 1, the better performance of LSSA is for the problem considered. For the random Ising problems with sizes from $N_p = 8$ to $N_p = 20$, the ground state and ground state energy of the full system and subsystems can be calculated by exact diagonalization in reasonable time, so both the full system and subsystem problems are solved

by exact diagonalization while the coefficient optimization part for the subsystems' contributions is done on `ibmq.qasm-simulator`. In this case the performance ratio is $R_p \equiv \frac{\text{LSSA GSE}}{\text{Exact GSE}}$. The corresponding result is shown in Fig. 3(a).

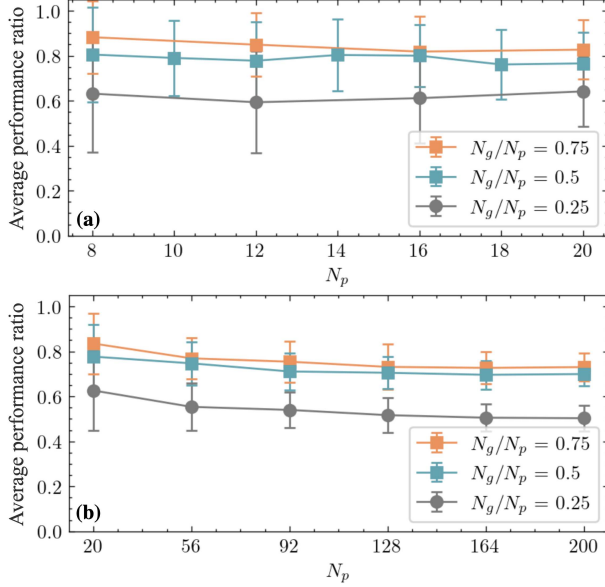


FIG. 3. (a) Average performance ratio for the random Ising problems with different values of N_p from 8 to 20. For each N_p , 100 random Ising Hamiltonians are tested and each is solved by sampling respectively size- $3N_p/4$, size- $N_p/2$ subsystems and size- $N_p/4$ subsystems $\lceil \frac{2N_p}{N_g} \rceil$ times. The subsystem problems are solved by exact diagonalization and the coefficient optimization is done by `ibmq.qasm-simulator`. The performance ratio is defined as $\frac{\text{LSSA GSE}}{\text{Exact GSE}}$ here. (b) Average performance ratio for the random Ising problems with different values of N_p from 20 to 200. Other parameters are similar to (a) except that the subsystem problems are solved by the `Dwave-tabu` classical solver and the performance ratio here is defined as $\frac{\text{LSSA GSE}}{\text{Dwave-tabu GSE}}$.

To explore the behavior of LSSA for problems with larger system sizes, from $N_p = 20$ to $N_p = 200$, we use the `Dwave-tabu` solver as our classical simulator, and the performance ratio in this case is defined as $R_p \equiv \frac{\text{LSSA GSE}}{\text{Dwave-tabu GSE}}$. The corresponding result in Fig. 3(b) shows how LSSA performs when sampling subsystems with sizes $N_g = 3N_p/4$, $N_g = N_p/2$ and $N_g = N_p/4$, randomly $N_s = \lceil \frac{2N_p}{N_g} \rceil$ times to solve the full problem with size N_p , compared to directly solving the size- N_p system by `Dwave-tabu`. As the sampling number of the subsystems is fixed to $N_s = \lceil \frac{2N_p}{N_g} \rceil$ for each N_p , so $N_s = 3$ for $N_g = 3N_p/4$, $N_s = 4$ for $N_g = N_p/2$ and $N_s = 8$ for $N_g = N_p/4$. Thus a smaller fraction of possible combinations of subsystems is sampled when the system size N_p is increased, i.e., only a fraction of $N_s / (\frac{N_p!}{N_g!(N_p-N_g)!})$ is sampled. So a decreasing trend in the average performance ratio can be seen in Fig. 3(b) when increasing the

system size N_p from 20 to 200. The average performance ratio, however, fluctuates a little bit with N_p as shown in Fig. 3(a), indicating the dependence on the spin variable configurations of the 4 (for $N_g = N_p/2$) or 8 (for $N_g = N_p/4$) randomly selected subsystems for smaller values of N_p from 8 to 20. By considering the Hilbert space dimension increases exponentially with the system (subsystem) size, even though we only sample quite a small fraction of the possible combinations of subsystems (e.g., $4 / (\frac{200!}{100!100!})$ for $N_p = 200$, $N_g = N_p/2$ and $N_s = 4$ subsystems), the average of the performance ratio with 100 random weighted Ising Hamiltonians can still reach $\approx 68\%$. One can also see from Figs. 3(a) and 3(b) that results for $N_g = 3N_p/4$ have a better average performance ratio than those for $N_g = N_p/2$ and $N_g = N_p/4$ as expected.

The effects of different subsystem sizes N_g and numbers of sampled subsystems N_s on the average performance ratio are investigated in Fig. 4. The subsystem sizes N_g of random Ising problems are tuned from a small size to the size of the full problem N_p in Fig. 4(a) for $N_p = 10$ and in Fig. 4(e) for $N_p = 20$ with N_s fixed to 4. As expected, the average performance ratio $R_p \rightarrow 1$ when $N_g \rightarrow N_p$ since more couplings between spins are considered when increasing N_g until the full problem is considered. For $N_p = 10$ in Fig. 4(b) and $N_p = 20$ in Fig. 4(f), we tuned N_s from a small number to the Hilbert space multiple with the size of a subsystem N_g fixed to the value of $N_p/2$. The Hilbert space multiple is defined as the multiple between the dimension of the full problem Hilbert space and that of the subsystem, so the Hilbert space multiple here is $2^{N_p}/2^{N_g} = 2^{N_p/2}$. We observed that the average performance ratio approaches a steady value and no longer increases when N_s is larger than some value that is quite smaller than the Hilbert space multiple. To be more specific, the average performance ratio ≈ 0.873 as $N_s \gtrsim 11$ for the $N_p = 10$ case and the average performance ratio ≈ 0.895 as $N_s \gtrsim 208$ for the $N_p = 20$ case. To investigate the effects of different values of N_g and N_s in larger system sizes N_p , we consider the cases for $N_p = 100$ and $N_p = 200$ with 10 random Ising problems tested for each data point of the average performance ratio with fixed values of N_g and N_s . Note that the definitions of the performance ratio in Figs. 4(a), 4(b), 4(e), and 4(f) is the same as that in Fig. 3(a), while in Figs. 4(c), 4(d), 4(g), and 4(h), it is the same as that in Fig. 3(b). That is, the denominator of the performance ratio in the former case is “Exact GSE” while it is “Dwave-tabu GSE” for the latter case. The behaviors of the average performance ratio as the subsystem size N_g is tuned up from a smaller value to the value of N_p with a fixed subsystem size number $N_s = 4$ for $N_p = 100$ and $N_p = 200$ are shown in Figs. 4(c) and 4(g), respectively. As expected, the behavior that the average performance ratio $\rightarrow 1$ as $N_g \rightarrow N_p$ is also observed. Similar to Fig. 4(f), N_g is fixed to $N_p/2$ for $N_p = 100$ in Fig. 4(d) and $N_p = 200$ in Fig. 4(h), while N_s is tuned up from 4 to 1024. Note that the data points

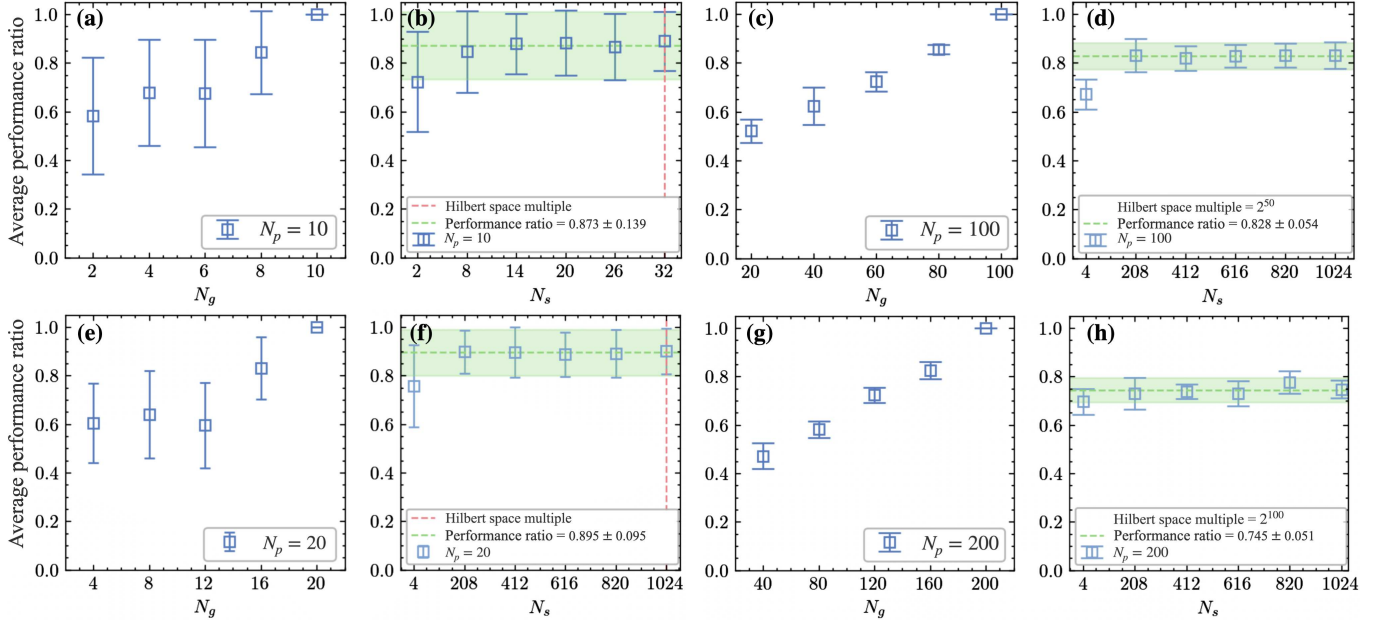


FIG. 4. Average performance ratio for the random Ising problems with different subsystem sizes N_g in (a), (e), (c) and (g), and different numbers of the sampled subsystems N_s in (b), (f), (d) and (h) for problem sizes $N_p = 10, 20, 100$ and 200 , respectively. In (a), (c), (e), and (g), N_s is fixed to 4, and in (b), (d), (f), and (h), N_g is fixed to $N_p/2$. For each data point in (a), (b), (e), and (f) 100 random Ising problems are used and in (c), (d), (g), and (h), 10 random Ising problems are used. The Hilbert space multiples $2^{N_p/2}$ are shown by the pink dashed line in (b) and (f), and are mentioned in values in the legends of (d) and (h).

are only up to $N_s = 1024$ as the data for N_s approaching the Hilbert space multiple in these cases is too large to be sampled (2^{50} and 2^{100}). Nevertheless, in the case of $N_p = 100$ in Fig. 4(d), the average performance ratio still converge to around 0.828 as $N_s \gtrsim 208$. In the $N_p = 200$ case, the average performance ratio grows very slowly with the average value of last 5 data points being 0.745, so we may assume that the average performance ratio will converge to some value with a sufficient large N_s as in previous cases. The results in Figs. 4(b), 4(d), 4(f), 4(h) show that the minimal N_s value requirement for LSSA to obtain a considerably good convergent performance for a fixed N_g value is quite small compared to the size of the Hilbert space of the full problem. To further increase the performance of the LSSA algorithm, one may increase the value of the subsystem size N_g .

In practical terms, the number of qubits of a quantum machine or computer is fixed, and thus the subsystem size to be solved by a quantum machine should also be fixed. It is crucial to find out how the performance of LSSA could be affected when solving different sizes of problems with fixed N_g . Considering the case that we only have access to a N_{gb} -qubit gate-based quantum computer, which can be used to solve both the subsystem problems with $N_g = N_{\text{gb}}$ and the coefficient optimization problems of the subsystems up to $N_s^{\text{max}} = 2^{N_{\text{gb}}}$. Again, this thought is tested by solving the subsystem problems with **Dwave-tabu** and then by using the **ibmq.qasm-simulator** to do the coefficient optimization part. Besides the fully-connected

random Ising problems, 3-regular random Ising problems are also tested. A 3-regular random Ising problem is the case that each spin at a site interacts with other 3 spins at random sites with random coupling strengths, and thus the number of couplings are significantly smaller than the fully-connected case. The ranges of the random couplings J_{ij} and biases h_i of the 3-regular random Ising problems are also both constrained in the range of $(-1, 1)$. The performance ratio here is defined as $R_p \equiv \frac{\text{LSSA GSE}}{\text{Dwave-tabu GSE}}$. The average performance ratios of the fully-connected random Ising problems (labeled as “Fully”) with $N_{\text{gb}} \in \{5, 7, 16\}$ and the 3-regular random Ising problems (labeled as “3-Regular”) with $N_{\text{gb}} = 5$ are tested for N_p from 8 to 160 (note that $160 = 5 \times 2^5$). The corresponding result is shown in Fig. 5(a), where 100 random problems are tested for each data point with $N_s = \lceil N_p/N_{\text{gb}} \rceil$. Interestingly, the results of the 3-regular random Ising problems are worse than those of the fully-connected cases. This may result from the subsystem sampling process that could select configurations with interactions (couplings) that do not exist in the chosen random Hamiltonian (since we only have 3 of those couplings for each spin site), and thus the LSSA approximation in this case acts differently from that of the fully-connected case. As shown in Fig. 5(a), for a fixed N_p , the average performance ratios with larger N_{gb} performing better is consistent with the results in Fig. 4, and the results are above 0.8 for $2N_{\text{gb}} \geq N_p$ in all of the fully-connected cases run on simulators. However, for a

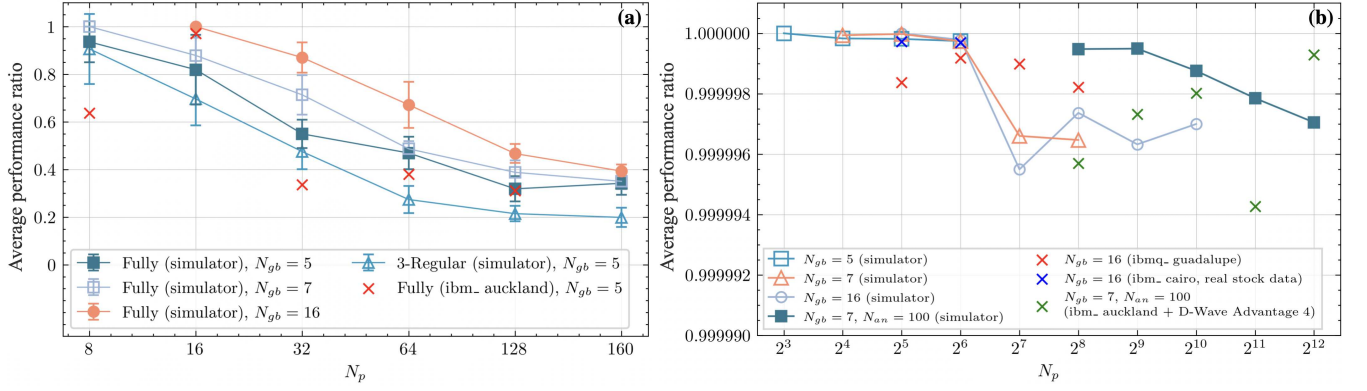


FIG. 5. (a) Average performance ratios for the fully-connected random Ising problems (labeled as “Fully”) with $N_{gb} = \{5, 7, 16\}$ and for the 3-regular random Ising problems (labeled as “3-Regular”) with $N_{gb} = 5$ and N_p from 8 to 160. We test 100 random problems for each data point by quantum simulators with $N_s = \lceil N_p / N_{gb} \rceil$ for each random Ising problem. For real hardware, a random Ising problem is tested for each N_p with $N_{gb} = 5$ and $N_s = \lceil N_p / 5 \rceil$. (b) Performance ratio for the portfolio optimization problems with N_p from 8 to 4096. The value of $N_{gb} = \{5, 7, 16\}$ indicates that $N_s = \lceil 2N_p / N_{gb} \rceil \leq N_s^{\max} = 2^{N_{gb}}$, and thus it is possible to use a single N_{gb} -qubit gate-based quantum chip for the problem with result labeled with the single value of N_{gb} . The label with $N_{gb} = 7$ and $N_{an} = 100$ means a hybrid structure such that the subsystem problems can be solved by an N_{an} -qubit annealing chip and the coefficient optimization can be done by an N_{gb} -qubit gate-based quantum computer, with $N_s = \lceil 2N_p / N_{an} \rceil \leq N_s^{\max} = 2^{N_{gb}}$.

given N_{gb} , the average performance ratio decreases with increasing N_p and becomes smaller than 0.5 for large N_p . In future work, we will develop a subsystem constructing method that depends on the coupling strengths between different sites, and this could improve the decreasing rate with N_p for the average performance ratio over the subsystem sampling scheme presented here. We will investigate in the next subsection the conditions of Ising problems for which LSSA could have rather good performance.

B. Conditions for good performance by LSSA

In the previous section, we have investigated the behavior of LSSA on the fully-connected random Ising problems described by Eq. (1) in the parameter regions where $J_{ij} \in (-1, 1)$ and $h_i \in (-1, 1)$. Since the distribution of the values of J_{ij} and h_i could affect the degree of difficulty to solve a corresponding Ising problem, we will investigate in what characteristic parameter regions that LSSA could be a good approximation. Let us define the

average values and standard deviations of J_{ij} and h_i as

$$\bar{J} = \frac{2}{N_p(N_p - 1)} \sum_{i>j} J_{ij}, \quad (12)$$

$$\bar{h} = \frac{1}{N_p} \sum_{i=1}^{N_p} h_i, \quad (13)$$

$$\text{std}(J) = \sqrt{\frac{2}{N_p(N_p - 1)} \sum_{i>j} (J_{ij} - \bar{J})^2}, \quad (14)$$

$$\text{std}(h) = \sqrt{\frac{1}{N_p} \sum_{i=1}^{N_p} (h_i - \bar{h})^2}. \quad (15)$$

Note that the factor in Eq. (12) is because there are $\frac{N_p(N_p-1)}{2}$ couplings J_{ij} for a fully-connected Ising problem with size N_p . In the previous section, the characteristic quantities of the distribution of the parameters are $\bar{J} = \bar{h} \approx 0$ and $\text{std}(J) = \text{std}(h) \approx 0.6$. In this section, we will discuss from the perspective of tuning the standard deviations and average values of J_{ij} and h_i the conditions to make LSSA perform well, and will test the derived conditions by investigating the performance ratio in the corresponding parameter regions of the Ising problems.

1. $\bar{J} > 0$

First, we start from the case that the average value of J_{ij} is larger than 0, i.e., $\bar{J} > 0$. Consider the following conditions: $\bar{J} \gg \text{std}(J)$, $|\bar{h}| > \text{std}(h)$, and $\bar{J} \gtrsim |\bar{h}|$.

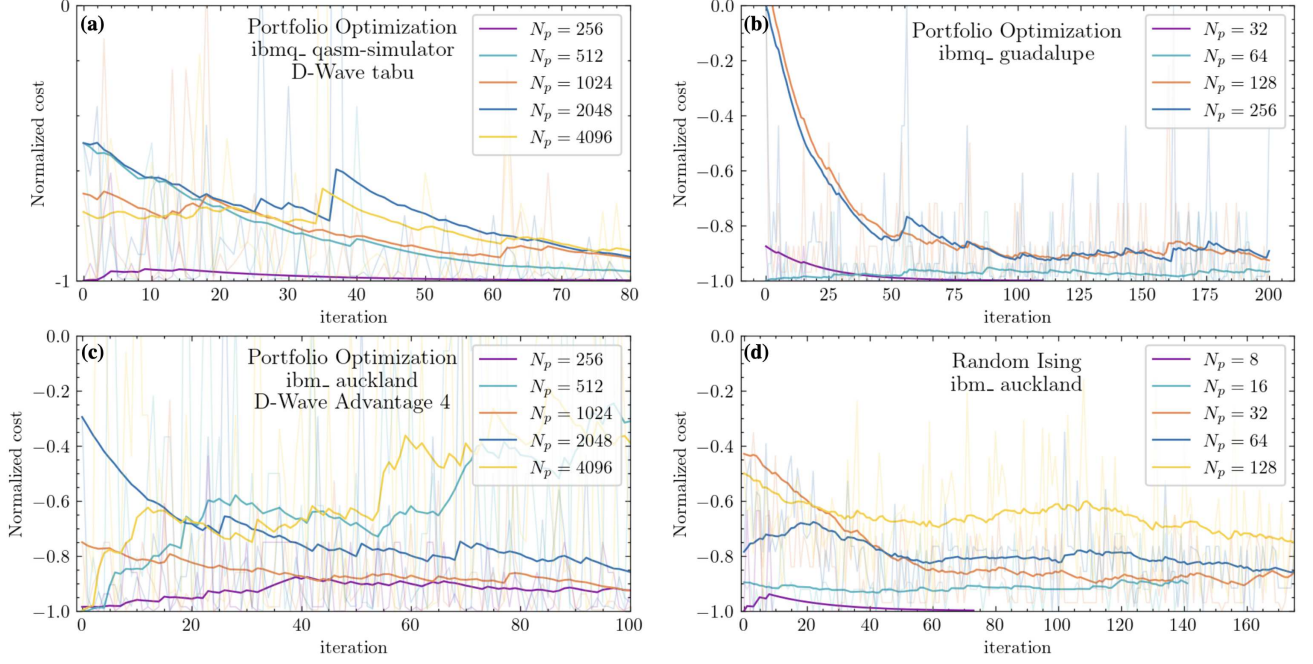


FIG. 6. Normalized cost during the optimization process for different cases: (a) Portfolio optimization with a hybrid computation structure calculated by simulators, corresponding to the solid purple square in Fig. 5(b). (b) Portfolio optimization with a gate-based computation structure calculated by `ibmq_guadalupe`, corresponding to the red crosses in Fig. 5(b). (c) Portfolio optimization with a hybrid computation structure calculated by `ibmq_auckland` and D-Wave Advantage 4, corresponding to the orange crosses in Fig. 5(b). (d) Random Ising problems gate-based computation structure calculated by `ibmq_auckland`, corresponding to the red crosses in Fig. 5(a). Data are smoothed in order to better observe the behavior, while the original data are faded in the background.

In such case, $J_{ij} \approx \bar{J}$ since $\text{std}(J)$ is relatively small compared to $\bar{J}$, and the expectation value of the problem Hamiltonian Eq. (1) can be approximately written as and roughly bounded by:

$$\langle H^d \rangle \lesssim \langle \bar{J} \sum_{i>j} \sigma_i^z \sigma_j^z + \sum_{i=1}^{N_p} h_i \sigma_i^z \rangle \lesssim \langle H^u \rangle, \quad (16)$$

where

$$\langle H^d \rangle = \langle \sum_i \sigma_i^z \bar{J} (\sum_j \sigma_j^z - 1) \rangle, \quad (17)$$

$$\langle H^u \rangle = \langle \sum_i \sigma_i^z \bar{J} (\sum_j \sigma_j^z + 1) \rangle. \quad (18)$$

This is because we could in this case consider the contribution from the coupling terms first and then the bias terms in the Hamiltonian. For $\bar{J} > 0$, the low energy states should be in a state similar to $|\uparrow\downarrow\uparrow\downarrow \dots \uparrow\downarrow\rangle$, as each of the two coupled spins with opposite orientations possesses a lower energy. Although there may be still quite a few frustrations, the number of frustrated spins is expected to be considerably smaller than the number of spins that are not frustrated. Consequently, the contributions from the terms in $\sum_{i=1}^{N_p} h_i \sigma_i^z$ will also counteract with one another, making $\langle \sum_{i=1}^{N_p} h_i \sigma_i^z \rangle$ small.

Moreover, in this case, neglecting the couplings between subsystems in LSSA may not make significant difference as the orientations of most of the coupling spins are alternately opposite with one another and thus most of the coupling terms are canceled out. A short summary in the case of $\bar{J} > 0$ is as follows: the conditions

$$\bar{J} \gg \text{std}(J), \quad |\bar{h}| > \text{std}(h) \quad (19)$$

$$\bar{J} \gtrsim |\bar{h}|. \quad (20)$$

could make LSSA perform well in the fully-connected random Ising problems.

In Figs. 7(a), (b), and (c), we investigate the performance ratio of LSSA for fully-connected random Ising problems with different values of the parameters $\text{std}(J)/\bar{J}$ and $|\bar{h}|/\bar{J}$ for $N_p = 64, 256$, and 1000 , respectively. The values of $\text{std}(J)/\bar{J}$ investigated are $\text{std}(J)/\bar{J} (\%) \in \{0.1, 1, 5, 10, 30, 50\}$, and those for $|\bar{h}|/\bar{J}$ are $|\bar{h}|/\bar{J} (\%) \in \{1, 10, 30, 50, 90, 110, 150, 200, 250\}$. For each combination of $\text{std}(J)/\bar{J}$ and $|\bar{h}|/\bar{J}$, we show the best result out of 10 attempts as a block in the figure. For Figs. 7, we set the parameter $\text{std}(h)/|\bar{h}|$ to be 10%. In Fig. 7(a), the problem size is $N_p = 64$, and the subsystem size is set to be $N_g = 5$, leading to $N_s = 2\lceil N_p/N_g \rceil = 26$. We observe that in the regime satisfying the conditions of Eqs. (19) and (20), which is the lower left part of the plot, the performance ratio is quite good (> 0.9)

compared to other regimes. The plot also shows that condition (19) is more restrict than condition (20) as when $|\bar{h}|/\bar{J}$ (%) > 100, the performance ratio can still be larger than 0.9. Similar behaviors can also be observed in Figs. 7(b) and (c), where for $N_p = 256$, $N_g = 7$, $N_s = 2\lceil N_p/N_g \rceil = 74$, and for $N_p = 1000$, $N_g = 16$, $N_s = 2\lceil N_p/N_g \rceil = 126$.

2. $\bar{J} < 0$

Next, we discuss the case when $\bar{J} < 0$. Consider the following conditions: $|\bar{J}| > \text{std}(J)$, $|\bar{h}| > \text{std}(h)$, and $|\bar{J}| \sim |\bar{h}|$, where the couplings and biases are distributed not far from the average value, and their signs are assumed to be almost the same as the signs of the average values. As a result, the contribution of the coupling terms dominate that of the bias terms in the Hamiltonian. In such a case, the coupling terms in the Hamiltonian (1) contribute a lower energy expectation value when the spins are in the same orientation, and the direction of the orientations of the spins is determined by the sign of $\bar{h}$. The task of finding the ground state becomes trivial, and thus the result constructed from subsystems can approximate the full system configuration well. In Fig. 7(d), we investigate the performance ratio of LSSA for $\bar{J} < 0$, $N_p = 64$, $N_g = 5$, and $N_s = 2\lceil N_p/N_g \rceil = 26$ for different parameters of $\text{std}(J)/\bar{J}$ and $|\bar{h}|/|\bar{J}|$. The values of $\text{std}(J)/\bar{J}$ investigated are $\text{std}(J)/\bar{J}$ (%) $\in \{1, 5, 10, 30, 50\}$, and those for $|\bar{h}|/|\bar{J}|$ are $|\bar{h}|/|\bar{J}|$ (%) $\in \{1, 10, 30, 50, 90, 110, 150, 200, 250\}$. The parameter $\text{std}(h)/|\bar{h}|$ is also fixed at 10% here. We observe that in this case, the performance ratio is almost 1 for the entire regime investigated. As a short summary in the case of $\bar{J} < 0$, the conditions

$$|\bar{J}| > \text{std}(J), |\bar{h}| > \text{std}(h), |\bar{J}| \sim |\bar{h}|, \quad (21)$$

and the signs of J_{ij} and h_i are almost the same as the signs of their average values could make LSSA perform well in the fully-connected random Ising problems. After the discussion of the good performance conditions, in the next section, a practical example of the problem lies within the conditions mentioned above will be shown.

C. Portfolio Optimization Problems

The random Ising problems are considered to be a general case of the problems we may encounter when dealing with quantum optimization problems. To investigate the applicability of LSSA to specific problems, we use portfolio optimization problems as our examples.

A portfolio optimization problem can also be formulated into a QUBO problem [6, 7], which can be further mapped to an Ising problem. We consider the portfolio optimization problem in the form [41]

$$H = -\mu^T \omega + \frac{\gamma}{2} \omega^T \Sigma \omega + \rho(1^T \omega - K)^2, \quad (22)$$

where $\omega \in \{0, 1\}^{N_p}$ is an N_p -dimensional vector of binary decision variables, $\mu \in \mathbb{R}^{N_p}$ is the expected return vector and $\Sigma \in \mathbb{R}^{N_p \times N_p}$ is the covariance matrix of size $N_p \times N_p$. The term $\mu^T \omega$ represents the return of the portfolio and $\frac{\gamma}{2} \omega^T \Sigma \omega$ denotes the risk term with non-negative parameter γ . The penalty coefficient ρ is to make sure that the constraint of the total number of investment being K can be satisfied. Here we also define volatility $\equiv \sqrt{\omega^T \Sigma \omega}$ and Sharpe ratio $\equiv \frac{\mu^T \omega}{\text{Volatility}}$. In this section, we use the simulated stock data to test LSSA on the portfolio problem, which consists of the prices of N_p assets over 30 days, with parameters $\gamma = 1$, $\rho = 10N_p$ and $K = N_p/2$. From the price data over D days, the expected return vector element μ_i is calculated by

$$\mu_i = \frac{1}{D-1} \sum_{d=1}^{D-1} \frac{p_{d+1}^i - p_d^i}{p_d^i}, \quad (23)$$

where p_d^i is the price of the i th asset of day d , and the covariance matrix element Σ_{ij} is

$$\Sigma_{ij} = \frac{1}{D-1} \sum_{d=1}^{D-1} \left(\frac{p_{d+1}^i - p_d^i}{p_d^i} - \mu_i \right) \left(\frac{p_{d+1}^j - p_d^j}{p_d^j} - \mu_j \right), \quad (24)$$

which represents the covariance between the return of asset i and asset j .

Next, to solve the portfolio optimization problem of Eq. (22), the mapping to the Ising model of Eq. (1) by using the relation $\omega_i = \frac{1-z_i}{2}$ is made and the corresponding results of h_i and J_{ij} are

$$h_i = \frac{1}{2} \mu_i + \rho K - \frac{1}{2} \rho N_p - \frac{\gamma}{4} \sum_{j=1}^{N_p} \Sigma_{ij}, \quad (25)$$

$$J_{ij} = \frac{1}{2} \left(\frac{\gamma}{4} \Sigma_{ij} + \rho \right), \quad (26)$$

respectively. We should mention that for both the simulated stock data used in this section and the real stock data that will be introduced later, the value of $\Sigma_{ij} \sim 10^{-3}$ to 10^{-4} , and $\mu_i \sim 10^{-1}$ to 10^{-2} . Thus, since ρ is a constant in the order of N_p , we can observe that $\bar{J} \sim \frac{1}{2} \rho$ is at least 10^5 order larger than $\text{std}(J) \propto \frac{\gamma}{8} \Sigma_{ij}$ if $N_p \geq 10$, implying $\bar{J} \gg \text{std}(J)$. Next, with the choice of $K = N_p/2$ (buy half of the stocks), the second and third terms of Eq. (25) cancels out each other, thus the condition $\bar{J} \gtrsim |\bar{h}|$ is satisfied since $\bar{J}$ is in the order of N_p and h_i is in the order of $\mu_i - N_p \Sigma_{ij}$. Although the condition $|\bar{h}| > \text{std}(h)$ is not checked intuitively, the quantity $\text{std}(h)/|\bar{h}|$ is observed to be below 50% in our calculations. We find that the portfolio optimization problems we test satisfies the good performance conditions Eqs. (19) and (20) of LSSA. This is why the performance of LSSA on the portfolio optimization problems could be expected to be quite good compared to that on the fully-connected random Ising problems.

The performance ratio versus full-problem size N_p for the portfolio optimization problems is shown in

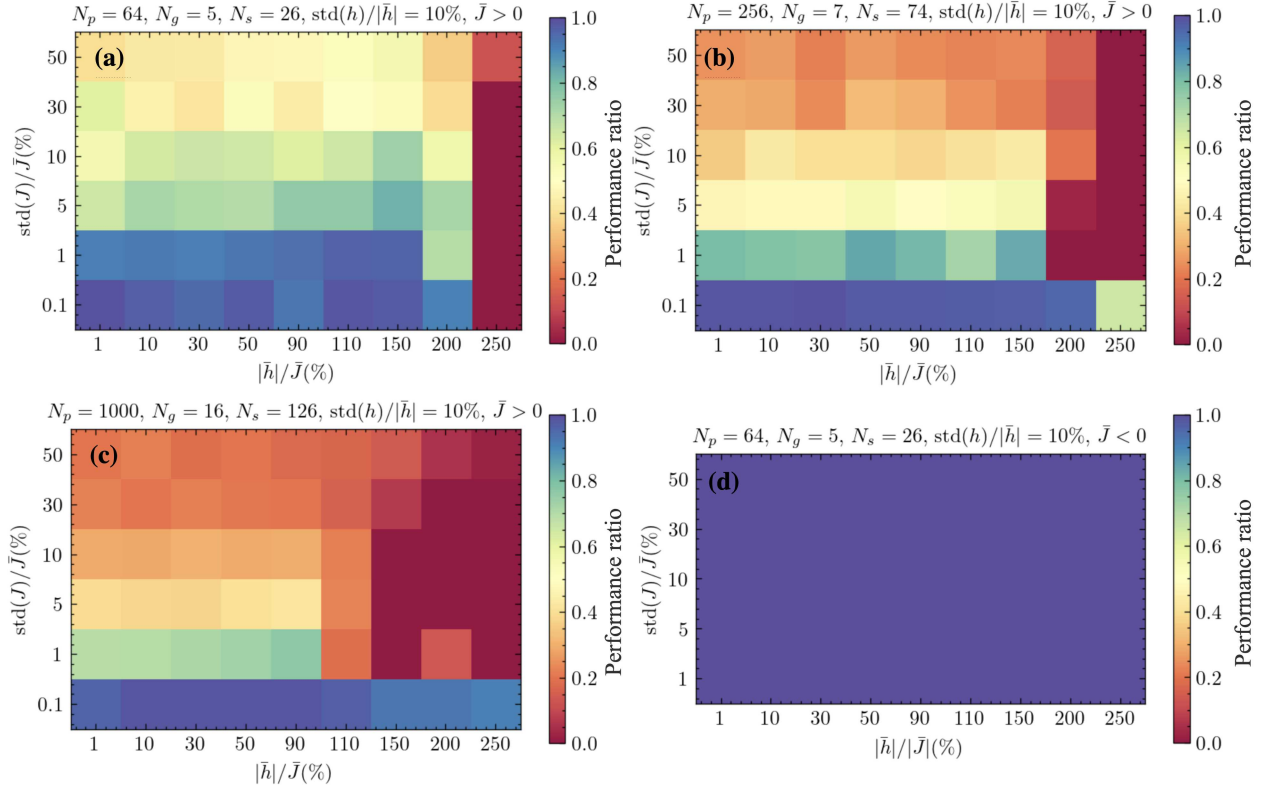


FIG. 7. Performance ratios of the fully-connected random Ising problem with different $\text{std}(J)/\bar{J}$, $|\bar{h}|/\bar{J}$, and $\text{sign}(\bar{J})$, where $\text{std}(h)/\bar{h}$ is fixed to 10%. For each data point, best of its 10 attempts is shown. (a) $\bar{J} > 0$ with $N_p = 64$, $N_g = 5$, $N_s = 2[N_p/N_g] = 26$. (b) $\bar{J} > 0$ with $N_p = 256$, $N_g = 7$, $N_s = 2[N_p/N_g] = 74$. (c) $\bar{J} > 0$ with $N_p = 1000$, $N_g = 16$, $N_s = 2[N_p/N_g] = 126$. (d) $\bar{J} < 0$ with $N_p = 64$, $N_g = 5$, $N_s = 2[N_p/N_g] = 26$. For (a), (b), and (c), the parameter investigated are $\text{std}(J)/\bar{J}$ (%) $\in \{0.1, 1, 5, 10, 30, 50\}$ and $|\bar{h}|/\bar{J}$ (%) $\in \{1, 10, 30, 50, 90, 110, 150, 200, 250\}$. For (d), the parameter investigated are $\text{std}(J)/\bar{J}$ (%) $\in \{1, 5, 10, 30, 50\}$ and $|\bar{h}|/\bar{J}$ (%) $\in \{1, 10, 30, 50, 90, 110, 150, 200, 250\}$.

Fig. 5(b). Here the performance ratio is defined as $R_p \equiv \frac{\text{LSSA GSE}}{\text{Dwave-tabu GSE}}$, measuring the performance with respect to the classical Dwave-tabu solver. Unlike the case of the random Ising problems for which we generate a lot of random instances and average their results, in the portfolio optimization, we deal with the same dataset and only consider different numbers of assets, which may make more sense if we consider to apply our method to the real stock market (see Sec. IV B 2). For each data point solved by simulators in Fig. 5(b), the best result of its 3 attempts is picked while the problem instance is fixed. By assuming only a 5-qubit gate-based quantum simulator (ibmq_qasm-simulator) available, the portfolio optimization problems with $N_p \in \{8, 16, 32, 64\}$ are tested using a single-layer QAOA with 5 iterations of the Nelder-Mead optimization to solve the subsystem problems, and the coefficient optimization process is the same as that in the random Ising case, where $N_s = \lceil 2N_p/5 \rceil$. The number of measurement shots for solving both the subsystem problems and the coefficient optimization are 8192. Interestingly, unlike the small average performance ratio obtained in Fig. 5(a) for the random Ising problems, LSSA holds up the performance ratio close to 1 pretty well for the portfolio optimization problems with $N_{\text{gb}} = 5$

shown in Fig. 5(b). The results show that the obtained cost function value is quite similar to that obtained directly by Dwave-tabu, with the performance ratio ≈ 1 , almost negligibly decreasing when scaling N_p up to 64. The effect of different numbers of measurement shots are also investigated in Appendix B. We find the results of the performance ratio shown in Fig. 14(d) for the portfolio optimization problems are not that sensitive to the numbers of measurement shots, and similar behavior for the random Ising problems can be observed in Fig. 14(c).

We also test $N_{\text{gb}} = 7$ for $N_p \in \{16, 32, 64, 128, 256\}$ and $N_{\text{gb}} = 16$ for $N_p \in \{32, 64, 128, 256, 512, 1024\}$ with $N_s = \lceil 2N_p/N_{\text{gb}} \rceil$ for the both cases, and the results are also shown in Fig. 5(b). Similar behavior of the performance ratio ≈ 1 is observed while a slightly larger decrease occurs for larger values of N_p . Note that $N_{\text{gb}} = \{5, 7, 16\}$ are selected based on the qubit numbers of the existing quantum computers provided by IBM Quantum. Next, as mentioned in Sec. II, it is possible to solve the subsystem problems with N_{an} -qubit annealing chip and optimizing the amplitudes of the solutions of the subsystem problems by N_{gb} -qubit gated-based quantum computer. We simulate this case with $N_{\text{gb}} = 7$ and $N_{\text{an}} = 100$ by solving the portfolio problems for

$N_p \in \{256, 512, 1024, 2048, 4096\}$ and $N_s = \lceil 2N_p/N_{\text{an}} \rceil$. For this case, the normalized cost, which is the cost function divided by the absolute value of the minimum cost, is shown in Fig. 6(a). We notice that the cost function saturates to a small value during the optimization process about 50 to 100 iterations, indicating that the initial state of the solved subsystem problems with random initialized coefficients constitutes a good initial guess.

IV. CALCULATIONS BY GATE-BASED QUANTUM COMPUTERS AND QUANTUM ANNEALERS

A. Random Ising Problems

After the investigation with the calculations performed by simulators, we use the gate-based quantum computers provided by IBM and the quantum annealer from D-Wave to perform the calculations to demonstrate the capability of the LSSA algorithm on the currently available quantum computing machines. More information about the quantum devices we use for the demonstrations can be found in Appendix D. For the subsystem problems solved by the gate-based quantum computers, a single layer QAOA is used with 5 iterations of the Nelder-Mead optimization, and 8192 measurement shots are used for each calculation of the expectation value. The results of the random Ising problem tested by `ibmq_auckland` with $N_g = N_{\text{gb}} = 5$ for $N_p \in \{8, 16, 32, 64, 128\}$ and $N_s = \lceil 2N_p/5 \rceil$ are shown in red crosses in Fig. 5(a), and the normalized cost in this case is shown in Fig. 6(d). In this section, the performance ratio is defined as $R_p \equiv \frac{\text{LSSA GSE}}{\text{Dwave-tabu GSE}}$ again. Although we only test one random Ising problem for each N_p , the trend of the average performance ratio is similar to that obtained by the simulators, i.e., it decreases considerably to a low value when $N_p \gg N_g$, indicating a relatively poor performance.

B. Portfolio Optimization Problems

1. Simulated Stock Data

First, the gate-based quantum computer of `ibmq_guadalupe` with $N_g = N_{\text{gb}} = 16$ is used to solve the portfolio optimization problems for $N_p \in \{32, 64, 128, 256\}$ and $N_s = \lceil 2N_p/16 \rceil$. The results are shown in red crosses in Fig. 5(b). Similar to the results by the simulator in Fig. 6(a), the normalized cost decreases to the lowest value in very few iterations as shown in Fig. 6(b). Then the hybrid annealing and gate-based computation structure is demonstrated by D-Wave Advantage 4 with $N_{\text{an}} = 100$ and `ibmq_auckland` with $N_{\text{gb}} = 7$ for solving the portfolio optimization problems for $N_p \in \{256, 512, 1024, 2048, 4096\}$ and $N_s = \lceil 2N_p/100 \rceil$. The results are shown in green crosses

in Fig. 5(b). Note that even though D-Wave Advantage 4 is a 5627-qubit system, the upper limit of the number of variables of an arbitrary (a fully-connected) Ising problem that can be solved by it is around 177 due to the connectivity geometry of the qubit hardware [42]. So we use this quantum annealing machine to conservatively solve the subsystem problems with size $N_{\text{an}} = 100$, and for each subsystem problem 5000 shots are used. The performance ratio in this case as shown in Fig. 5(b) is also almost equal to 1, which is quite impressive considering we only use $7 + 100$ qubits to solve the problem with size almost 40 times larger than the qubit number of our hardware. However, due to our definition of the performance ratio used here, the good performance just means the result is very close to that obtained from the `Dwave-tabu` solver, and is not guaranteed to be close to the actual solution of the problem.

2. US Stock Market Data

To examine the capability of LSSA for portfolio optimization in real world, we also test the real data from the US stock market [43] with a gate-based machine for subsystem problem size $N_{\text{gb}} = 16$ to solve the portfolio optimization problems of $N_p \in \{32, 64\}$, and $N_s = \lceil 2N_p/16 \rceil$. For each problem, we pick stocks of N_p largest Companies by market cap and use data over 47 months to construct our problem Hamiltonian, where parameters γ , ρ and K are the same as the above cases. The US stock data for $N_p = 32$ and $N_p = 64$ are shown in Figs. 8(a) and 8(d), respectively. Both the subsystem solving part and the solution coefficient optimization are done by the IBM Quantum machine `ibmq_cairo`. The result of the performance ratio is shown in blue crosses in Figs. 5(b). The Sharpe ratios obtained by both the classical solver (`Dwave-tabu`) and LSSA with `ibmq_cairo` for $N_p = 32$ and $N_p = 64$ shown in Fig. 8(b) and 8(c), respectively, are significantly larger than those from random stock portfolio compositions [scattered dots in Figs. 8(b) and 8(c)], but the Sharpe ratios from LSSA are slightly lower than those from the classical solver in both $N_p = 32$ and $N_p = 64$ cases. One may wonder why the points of both the optimized solutions from the classical solver and LSSA do not fall within but instead are far from the random sampling points (scattered dots) of the stock portfolio compositions. This is because the number of possible portfolio compositions that satisfy our constraint is $C(64, 32) = \frac{64!}{32!32!} \approx 1.8 \times 10^{18}$, and the number of random sampling portfolio compositions of 5×10^3 is only a very tiny portion of the full solution space. We show in Fig. 9 that with a larger number of random portfolio compositions of 5×10^6 , the distribution of the scattered points becomes wider. As a result, our optimized solutions becomes closer to the distribution of the results from the random portfolio compositions. Even though the number of random sampling points increases by a fac-

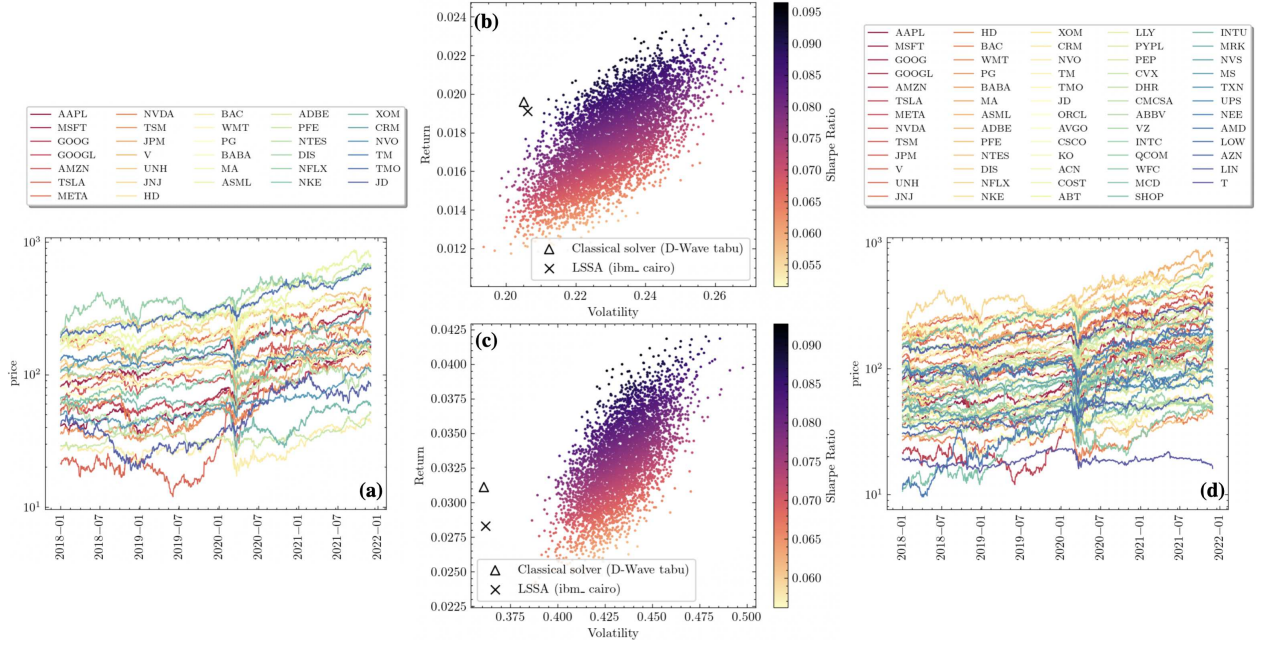


FIG. 8. Data and results of the Sharpe ratio of the the portfolio optimization problem for the US stock market. (a) US stock market data for the stocks of $N_p = 32$ largest companies sorted by market cap. (b) Return and volatility of the Sharpe ratios of the portfolio optimization problem with $N_p = 32$ stocks, obtained from the classical solver (triangle) and LSSA (cross). The scattered dots represent the results from random sampling of 5×10^3 combinations of the portfolios. (c) Similar to (b) but for $N_p = 64$ stocks. (d) Similar to (a) but for $N_p = 64$ stocks.

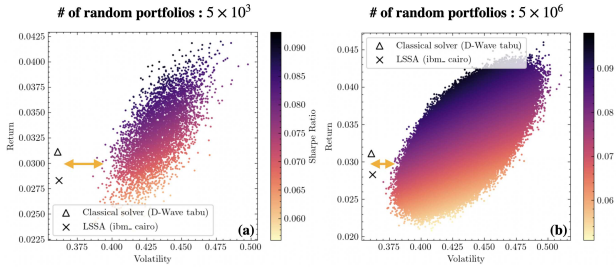


FIG. 9. Return and volatility for different numbers of random portfolio compositions (a) 5×10^3 and (b) 5×10^6 . The distance between the results obtained from the classical solver (triangle) and LSSA (cross) and the results from the random sampling portfolio compositions is indicated by an orange double-arrow line.

tor of 10^3 , it is still much smaller than the number of total possible portfolio compositions in the solution space. This also highlights the difficulty of trying to find the optimal solution of the combinatorial optimization problem through the random sampling search or the brute-force or exhaustive search algorithm. It is thus reasonable that the optimized solutions are distant from the distribution of the results obtained from the random portfolio compositions.

V. COMPUTATIONAL PROPERTIES OF LSSA

From the previous sections, we have demonstrated the performance ratio of LSSA for solving some Ising problems. However, the properties related to the computational time is also important when discussing a new method like LSSA. In this section, after describing the design of the metric to evaluate the computational time for LSSA, we compare the computational time between the Tabu solver and LSSA with different parameters and settings.

A full task computed on real quantum device(s) could be extremely time consuming because of the limited access to the shared devices as the tasks need to be in a queue for uncertain amount of time (from few minutes to few days). To estimate and compare the computational time of LSSA on real quantum devices with other approaches, this queue time need to be ignored to have a fair comparison. For estimating the LSSA computational time without the queue time, the execution time could be formulated as a function of the unit circuit execution time, so the estimation could be made once one obtains the unit circuit execution time from the average value of some sample circuits' execution times. As a result, the required execution time of LSSA for a Ising problem with size N_p , subsystem size N_g , and the number of subsys-

tems N_s on the gate-based chip can be formulated as

$$t_{\text{LSSA}}^{\text{gb}} = N_{\text{iter}}^{\text{sub}} \frac{N_s}{P_f} (t_{\text{gb}}^{\text{sub}} + t_{\text{cost}}^{N_g}) + N_{\text{iter}}^{\text{ao}} (t_{\text{gb}}^{\text{ao}} + t_{\text{cost}}^{N_p}). \quad (27)$$

In Eq. (27), the first term corresponds to the time needed to solve the N_s subsystem problems using QAOA with $N_{\text{iter}}^{\text{sub}}$ iterations. For each iteration, a quantum circuit execution time $t_{\text{gb}}^{\text{sub}}$ and a size- N_g classical cost function evaluation time $t_{\text{cost}}^{N_g}$ are required. The symbol P_f denotes the number of quantum processor units (QPUs) that are used in parallel to solve the subsystem problems. The second term corresponds to the coefficient optimization part with $N_{\text{iter}}^{\text{ao}}$ iterations, in which each iteration includes a quantum circuit execution time $t_{\text{gb}}^{\text{ao}}$ of the VQE and a size- N_p classical cost function evaluation time $t_{\text{cost}}^{N_p}$.

Note that since the qubit usage for the coefficient optimization part is $N_{\text{gb}} = \lceil \log_2 N_s \rceil$, in the case that $N_s = \lceil N_p / N_g \rceil < 2^{N_{\text{gb}}-1}$, the actual value of $t_{\text{gb}}^{\text{ao}}$ could be less than or equal to $t_{\text{gb}}^{\text{sub}}$ as the coefficient optimization part can be executed with fewer qubits. Similar to the case for the gate-based chip, the required execution time for LSSA in a hybrid gate-based and annealing scheme is

$$t_{\text{LSSA}}^{\text{an}} = \frac{N_s}{P_f} (t_{\text{an}}^{\text{sub}} + t_{\text{cost}}^{N_g}) + N_{\text{iter}}^{\text{ao}} (t_{\text{gb}}^{\text{ao}} + t_{\text{cost}}^{N_p}), \quad (28)$$

where $t_{\text{an}}^{\text{sub}}$ is the time for the quantum annealer to solve a subsystem problem. The upper limit of $t_{\text{an}}^{\text{sub}}$ is restricted to 1 s in the cloud quantum annealer platform that we used for the calculations, and thus we set $t_{\text{an}}^{\text{sub}} = 1$ s in the following comparison. For the $N_g = 16$ case, we use $t_{\text{gb}}^{\text{sub}} = t_{\text{gb}}^{\text{ao}} = 10.09$ s, the average value of the circuit execution time of `ibm_cairo` with 8192 shots, in Fig. 5(b). The values of $t_{\text{cost}}^{N_g}$ and $t_{\text{cost}}^{N_p}$ of sizes N_g and N_p , respectively, are evaluated by the classical computer. With these, the values of $t_{\text{LSSA}}^{\text{gb}}$ and $t_{\text{LSSA}}^{\text{an}}$ can be obtained for each parameter setting.

In Fig. 10, the estimated execution times for LSSA in the pure gate-based case with $N_{\text{gb}} = 16$, $N_s = \lceil N_p / 16 \rceil$, $N_{\text{iter}}^{\text{sub}} = 20$, $N_{\text{iter}}^{\text{ao}} = 200$, with 2 different parallel factors, $P_f = 1$ and $P_f = 4$, are shown in orange triangles and orange squares, respectively. Those in the hybrid gate-based and annealing case with $P_f = 1$, $N_{\text{an}} = 100$, $N_{\text{gb}} = 16$, $N_s = \lceil N_p / 100 \rceil$, with 2 different iterations of the coefficient optimization, $N_{\text{iter}}^{\text{ao}} = 200$ and $N_{\text{iter}}^{\text{ao}} = 50$, are shown in dark orange hollow circles and dark green triangles, respectively. A hypothetical case of the quantum circuit execution time $t_{\text{gb}} = 0.1$ s is also shown in light green hollow circles, which denotes a faster circuit execution time that may be achieved in the near future. As a comparison, the execution times for simulated annealing, steepest descent, and tabu solver, executed on a classical processor (Intel i7), are depicted as blue triangles, dark orange triangles, and dark purple squares, respectively. For the results by real quantum devices in Sec. IV, the hybrid gate-based and annealing

(`ibm_auckland` + D-Wave Advantage 4) case is shown in dark blue crosses for $N_p \in \{256, 512, 1024, 2048, 4096\}$, and the pure gate-based case by `ibm_guadalupe` is plotted in dark green cross for $N_p = 256$. Note that in these cases, the queue time for executing the computation is ignored in order to have the computational time that fully depends on the LSSA calculation.

The execution time in Eq. (27) and Eq. (28) for the gate-based scheme and hybrid gate-based and annealing scheme, respectively, can be divided into two parts: time on classical hardware (classical hardware time) and time on quantum hardware (quantum hardware time). The classical hardware times representing the evaluation of the cost functions in both equations are given respectively by:

$$t_{\text{LSSA, cl}}^{\text{gb}} = N_{\text{iter}}^{\text{sub}} \frac{N_s}{P_f} t_{\text{cost}}^{N_g} + N_{\text{iter}}^{\text{ao}} t_{\text{cost}}^{N_p}, \quad (29)$$

$$t_{\text{LSSA, cl}}^{\text{an}} = \frac{N_s}{P_f} t_{\text{cost}}^{N_g} + N_{\text{iter}}^{\text{ao}} t_{\text{cost}}^{N_p}. \quad (30)$$

On the other hand, the quantum hardware times involve solving subsystems and coefficient optimization, and are given respectively by:

$$t_{\text{LSSA, qu}}^{\text{gb}} = N_{\text{iter}}^{\text{sub}} \frac{N_s}{P_f} t_{\text{gb}}^{\text{sub}} + N_{\text{iter}}^{\text{ao}} t_{\text{gb}}^{\text{ao}}, \quad (31)$$

$$t_{\text{LSSA, qu}}^{\text{an}} = \frac{N_s}{P_f} t_{\text{an}}^{\text{sub}} + N_{\text{iter}}^{\text{ao}} t_{\text{gb}}^{\text{ao}}. \quad (32)$$

To demonstrate the relationship between the classical hardware time and the quantum hardware time in the LSSA calculation as a function of the problem size N_p , we introduce the percentage of classical hardware time, denoted as $p_{\text{gb}} \equiv \frac{t_{\text{LSSA, cl}}^{\text{gb}}}{t_{\text{LSSA}}^{\text{gb}}} \times 100\%$ for the pure gate-based scheme and $p_{\text{an}} \equiv \frac{t_{\text{LSSA, cl}}^{\text{an}}}{t_{\text{LSSA}}^{\text{an}}} \times 100\%$ for the hybrid gate-based and annealing scheme. The classical hardware time percentage provides information about both the classical hardware time and quantum hardware time components, as the total execution time of LSSA, either $t_{\text{LSSA}}^{\text{gb}}$ or $t_{\text{LSSA}}^{\text{an}}$, is the sum of these two components. In Fig. 11, we present the classical hardware time percentage with data from Fig. 10 using the same plot and legend marker style, enabling a better comparison between the two figures and a clearer understanding of the relationship.

Compare to the classical tabu solver, LSSA may reach a better computational time in the case that the calculation of the coefficient optimization converges in less iterations (about 50 to 100 iterations) and in the case with a larger problem size ($N_p > 4000$) for the hybrid gate-based and annealing architecture (dark triangles in Fig. 10). The computational time could be further reduced if the quantum circuit execution time is reduced in the near future (light green hollow circles in Fig. 10). Thus, with the potential advantages in the computational time, LSSA makes the problem sizes of certain problems that can be solved by the current quantum hardware comparable to that by the classical solvers.

we next discuss and compare the computation time complexity of LSSA with the conventional VQE approach. It is important to note that the conventional VQE, which tackles problems of size- N_p on N_p -qubit hardware, is unable to manage large-scale problems due to the limitation of currently available quantum resources, such as the number of qubits and the depth for quantum circuit for reliable quantum computation. In comparison, LSSA partitions the full-size problem into smaller subsystem problems enabling it to solve a much larger-size problem than conventional VQE, and even though LSSA in this case needs to solve many subsystem problems, the scaling behavior of its computational time is similar to that of the conventional VQE approach. To show this, we first estimate the computational time of the conventional VQE as

$$t_{\text{VQE}} = N_{\text{iter}}(t_{\text{gb}} + t_{\text{cost}}^{N_p}), \quad (33)$$

where t_{gb} is the size- N_p quantum circuit execution time for the VQE and $t_{\text{cost}}^{N_p}$ is the function evaluation time for the size- N_p classical cost function. The computation overhead of LSSA is just the first term in Eq. (27) and Eq. (28), which grows linearly in N_p because

$$\frac{N_{\text{iter}}^{\text{sub}}}{P_f}(t_{\text{gb}}^{\text{sub}} + t_{\text{cost}}^{N_g}) = \text{constant}, \quad \text{and} \quad (34)$$

$$\frac{t_{\text{an}}^{\text{sub}} + t_{\text{cost}}^{N_g}}{P_f} = \text{constant}, \quad (35)$$

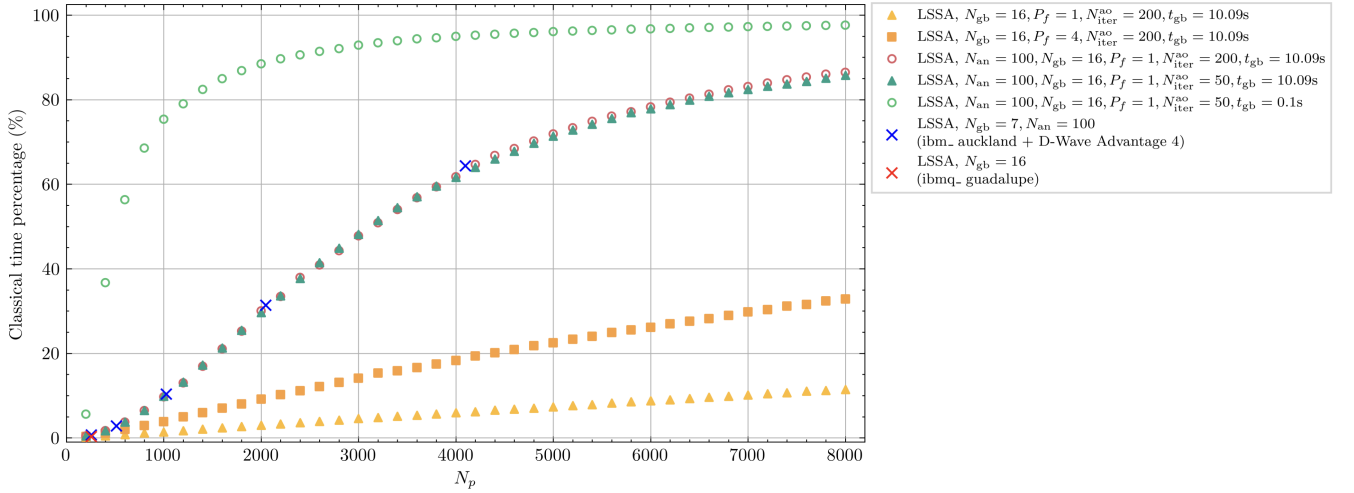
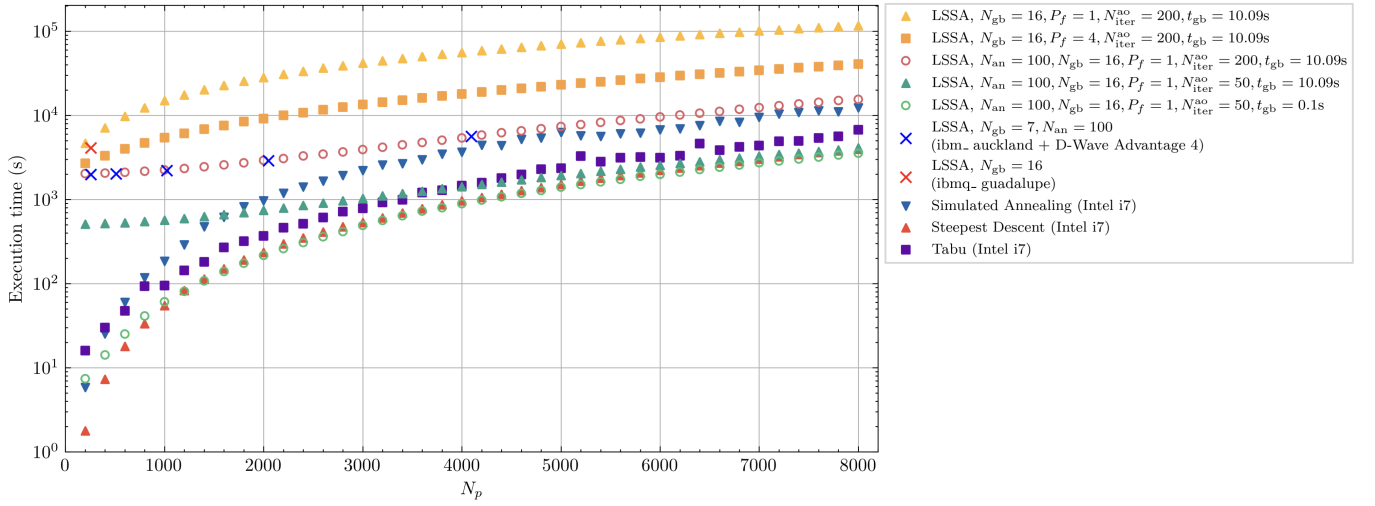
with respect to N_p , and $N_s = k \lceil \frac{N_p}{N_g} \rceil \propto N_p$ for some integer k . The values of Eq. (34) and Eq. (35) can be reduced by increasing the number of QPUs, P_f , to compute in parallel. Note that in general $t_{\text{cost}}^{N_p}$ grows quadratically in N_p since the number of terms in the fully-connected Ising Hamiltonian grows quadratically in N_p . Therefore, $t_{\text{cost}}^{N_p}$ is the dominant term in both Eq. (27) and Eq. (28), which implies that LSSA can reduce the qubit usage while keeping the computational complexity quadratic as the conventional VQE as shown in Eq. (33). With the possibility to significantly reduce the qubit usage and increase the reliable circuit depth, and the potential of faster computational time, LSSA could be useful compared to both the classical solver and the conventional VQE.

VI. DISCUSSIONS

Although the idea to reduce the required number of qubits to solve a larger-size problems seems compelling, the fact that the subsystem sampling method of LSSA neglects some of the couplings between variables (spin sites) makes it not applicable to all kinds of Ising problems. For the random Ising problems, the average performance ratio (see Fig. 3 and Fig. 4) could be increased by tuning up N_g and N_s , while its scaling with N_p is in a

declining trend. In this case, LSSA could be useful when $(N_p/N_g) \sim O(10^0)$, i.e., when the number of qubits of the quantum machine is not far from the size of the target problem. If $(N_p/N_g) \sim O(10^1)$, we find in Fig. 5(a) that the problem size N_p is too large to have a reasonably good solution for the random Ising problems by LSSA using both the quantum simulator and real hardware (ibm_auckland). However, when it comes to the portfolio optimization problems, even when $N_p/N_g \sim O(10^1)$, the average performance ratio can be very close to 1, maintaining in the range of $1 - O(10^{-6})$ using both the simulator and real hardware (ibmq_guadalupe, ibm_auckland, ibm_cairo and D-Wave Advantage 4). This could result from the coupling strengths are quite different for the random Ising problems and portfolio optimization problems. The strengths of coupling J_{ij} and biases h_i of the random Ising problem are distributed randomly between the values -1 and 1 , so their strengths are comparable. While for the portfolio optimization problem, due to the magnitude of the expected return and the covariance calculated from both the simulated and real stock data, the distributions of coupling strengths and the biases satisfy the good performance condition mentioned in Sec. III B. Thus the neglected couplings in the subsystem sampling process do not contribute significantly to the obtained full-system ground state energy for the portfolio optimization problem as compared with the random Ising problem. Furthermore, the VQE subsystem coefficient optimization procedure with the full-system Hamiltonian in the final stage also tries to adjust and recover some correlation lost in the subsystem sampling. This also helps LSSA have a very high-quality performance ratio for the portfolio optimization problems as the correlation lost due to the subsystem sampling process is not that substantial.

The maximum number of N_s is determined by the size of the gate-based quantum computer N_{gb} for coefficient optimization of the solution of the subsystem problems, i.e., $N_s^{\text{max}} = 2^{N_{\text{gb}}}$. Therefore, the problem size N_p could be scaled up to $N_p^{\text{max}} = N_{\text{gb}} N_s^{\text{max}} = N_{\text{gb}} 2^{N_{\text{gb}}}$ for a gate-based machine of size N_{gb} , i.e., each spin site should be at least sampled once. Since there is no causality between solving different subsystem problems in LSSA, it is possible to solve the subsystem problems in parallel if we have multiple quantum computers of the same or larger sizes. On the other hand, one can in a hybrid structure take the advantage of a larger effective number of qubits of quantum annealing machines to solve subsystem Ising problems of a larger size (i.e., $N_{\text{an}} > N_{\text{gb}}$), and then can use a gate-based quantum computer to encode and optimize the subsystem solution coefficients in the Hilbert space to find the solution of a full-system problem of a larger size up to $N_p^{\text{max}} = N_{\text{an}} 2^{N_{\text{gb}}}$. Alternatively, one can also take the advantage of $N_{\text{an}} > N_{\text{gb}}$ to decrease the number of N_s to $\tilde{N}_s = \frac{N_{\text{gb}}}{N_{\text{an}}} N_s$ when keeping the full-system-problem size fixed. For the result demonstrated in Fig. 5, we do not solve the full-system problem of the maximum size N_p^{max} . Instead, we only solve prob-



lems of size $N_p < N_p^{\max}$ with the number of the subsystems $N_s < N_s^{\max}$ in order to have reasonable execution times to demonstrate the behavior of LSSA by our current hardware. This could be conquered by solving the subsystem problems in parallel as mentioned above if we

would have many small-size quantum machines.

So far, we have shown that a larger-size problem can be approximately solved by dividing it into problems of small-size subsystems. We call this LSSA a level-1 approximation as only one level of subsystem reduction is

made. However, this procedure is not constrained to be only done once. To further extend the concept of LSSA, we can even divide the subsystems into further smaller-size sub-subsystems, which may be useful when multiple quantum computers with a small system size are available, such that all sub-subsystem problems can be calculated in parallel. We called this case a level-2 approximation. In Appendix C, we show how a 5-qubit gate-based quantum computer is possible to approximately solve for the portfolio optimization problem with up to 5120 assets (stocks), yet still yielding pretty good performance.

VII. CONCLUSION AND OUTLOOK

Targeting to tackle the Ising problems of a size much larger than that of our current hardware, we propose a new algorithm, LSSA, to sample the full-system problem into a number of smaller-size subsystems. The subsystem problems are then solved by the subsystem solvers of gate-based or annealing quantum machines. After that, the contributions of the solutions of different subsystems are considered as amplitudes in a VQE optimization algorithm using the full-system Hamiltonian, and the optimal contribution of each subsystem is then reconstructed to determine the approximate full-problem ground state configuration. We solve the random Ising problems with sizes up to 160 variables using 5 qubits in a gate-based quantum computer, and solve the portfolio optimization problems with sizes up to 4096 variables using 100 qubits in a quantum annealer and 7 qubits in a gate-based quantum computer using the level-1 approximation of LSSA. Our result gives an example to scale up the capability of the current quantum computing resources by combining different computational architectures of quantum machines. We also show the extendibility of LSSA to a deeper level of approximation by employing the level-2 approximation of LSSA for the portfolio optimization problems up to 5120 ($N_{\text{gb}}2^{2N_{\text{gb}}}$) variables with pretty good performance ratio values using only 5 (N_{gb}) qubits in a gate-based quantum computing scenario. The portfolio optimization problems could be excellent problems for LSSA as the exact ground-state solutions are desired but not required as long as for each problem, high-quality solutions close to the ground-state solution can be found. After all, one may not expect to have a portfolio composition that can give the maximum return and the minimum risk for every investment, but will be happy as long as one can make a substantial profit. Although the quality of the performance of LSSA could be problem-dependent, the future work aims to find what kinds of problems are suitable for LSSA and to improve the subsystem sampling effectiveness and efficiency by considering a coupling-dependent subsystem grouping method.

CODE AVAILABILITY

We provide the source code for the application of LSSA in fully-connected random Ising problems at [https://](https://github.com/CYLphysics/LSSA)

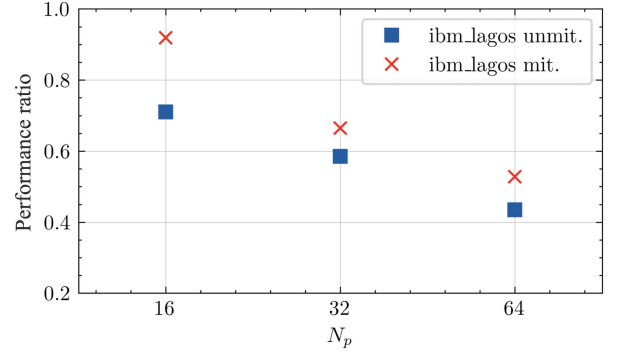


FIG. 12. The effect of readout error mitigation (REM) on the performance ratio for the fully-connected random Ising model solved by the LSSA algorithm was investigated. For problem size $N_p \in \{16, 32, 64\}$, the parameters N_g and N_s of the LSSA algorithm were set to 7 and $\lceil 2N_p/N_g \rceil$, respectively, and the results were obtained from the 7-qubit `ibm.lagos`.

github.com/CYLphysics/LSSA.

ACKNOWLEDGMENTS

We thank IBM Quantum Hub at NTU for providing computational resources and accesses for conducting the gate-based quantum computer demonstrations and the Quantum Technology Cloud Computing Center, National Cheng Kung University, Taiwan for providing accesses to Amazon Braket. We also acknowledge the support from MediaTek Inc., Taiwan. H.-S.G. acknowledges support from the National Science and Technology Council, Taiwan under Grants No. NSTC 112-2119-M-002-014, No. NSTC 111-2119-M-002-006-MY3, No. NSTC 111-2119-M-002-007, No. NSTC 110-2627-M-002-002, No. NSTC 111-2627-M-002-001, and No. NSTC 111-2627-M-002-006, from the US Air Force Office of Scientific Research under Award Number FA9550-23-S-0001, and from the National Taiwan University under Grant No. NTU-CC-112L893404. H.-S.G. is also grateful for the support from the “Center for Advanced Computing and Imaging in Biomedicine (NTU-112L900702)” through The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE), Taiwan, and the support from the Physics Division, National Center for Theoretical Sciences, Taiwan.

Appendix A: Effect of error mitigation

We briefly discuss the error and how to mitigate the error of the current quantum hardware. The current available quantum computing devices are known as noise intermediate-scale quantum (NISQ) hardware, which have a non-negligible effect of noise, resulting errors in the computation. Several error mitigation meth-

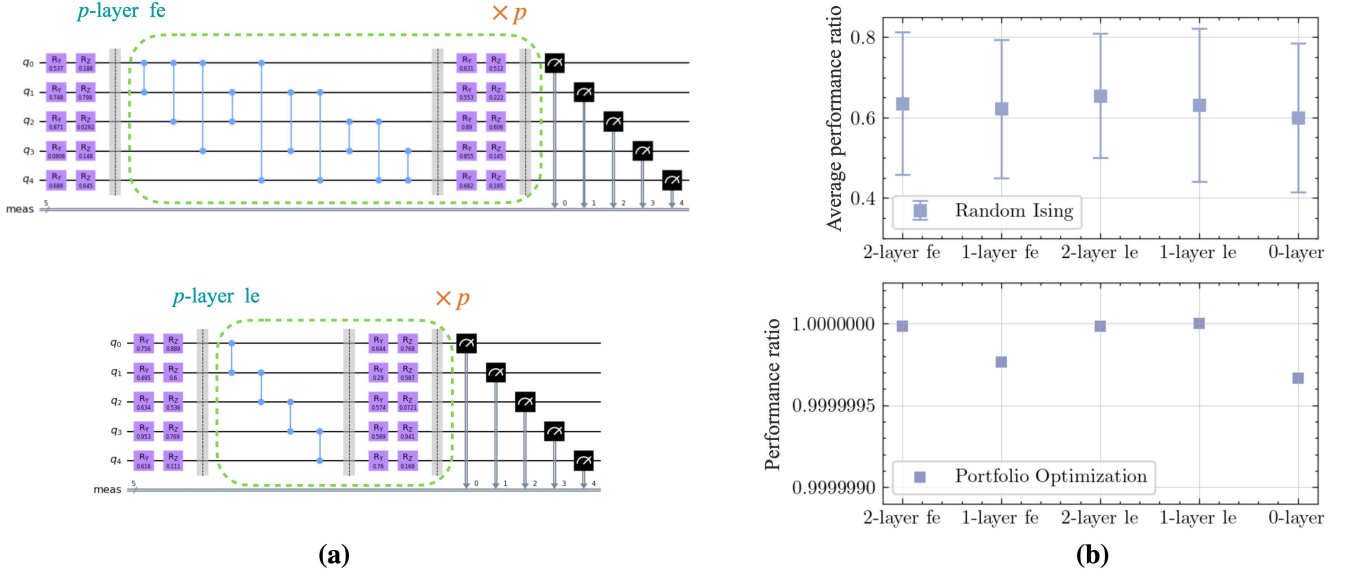


FIG. 13. Effects of different VQE circuit structures (ansatzes) for the subsystem coefficient optimization. (a) Visualization of the VQE circuits. The p -layer structure means that the circuit in the dashed box is repeated p times, and labels “fe” and “le” indicate the entanglement layer being full entanglement (the upper graph) and linear entanglement (the lower graph), respectively. Note that 0-layer means the circuit without the dashed box. (b) Average performance ratio of the random Ising problems and performance ratio of the portfolio optimization problems with $N_p = 20$, $N_g = 5$ and $N_s = 8$. For each data point of the random Ising problems, 100 problems are solved and averaged. In the portfolio optimization plot, the best result of their 10 attempts for the same problem is picked.

ods have been proposed to improve the result from these hardware, including readout error mitigation (REM) [44, 45], zero noise extrapolation (ZNE) [46], and probabilistic error correction (PEC) [46]. While REM may be effective in mitigating readout (measurement) errors, the noise in the quantum circuit remains unmitigated; however, the computational overhead is relatively minor. ZNE may have a better effect on noise mitigation, but the function form of the noise, which is unknown, is crucial for the extrapolation step, and the computational overhead is greater than that of REM. PEC is the most powerful method for dealing with noise, but the computational overhead is also significant, which could potentially negate the computational advantage of quantum computing. For the IBM hardware we use, the dominating error is the readout error, which can be handled well by REM. After weighing the pros and cons, we found that the REM was the most suitable option for our needs due to its efficiency and cost-effectiveness, thus we chose the REM to apply to our results.

In the context of REM [44, 45], consider a noisy measurement involving n qubits characterized by a positive operator-valued measure (POVM) with 2^n elements $\{\Pi_x\}$, where each element is labeled by an n -bit string x . Let ρ denote the output state of a noisy quantum circuit acting on these n qubits. The probability of observing a measurement outcome x is then expressed as $P(x) = \text{Tr}(\rho\Pi_x)$. Ideally, Π_x equates to $|x\rangle\langle x|$, but in the presence of noise, Π_x may represent any positive semi-definite operator satisfying the normalization condition

$\sum_x \Pi_x = I$. Assuming that all Π_x elements are diagonal in the computational basis facilitates more efficient error mitigation. Consequently,

$$\Pi_y = \sum_x \langle y|A|x\rangle|x\rangle\langle x| \quad (\text{A1})$$

for a noise-associated stochastic matrix A of dimension 2^n . Here, $\langle y|A|x\rangle$ signifies the probability of recording an outcome y when the actual outcome is x . For instance, with $n = 1$ and

$$A = \begin{bmatrix} 0.95 & 0.23 \\ 0.05 & 0.77 \end{bmatrix}, \quad (\text{A2})$$

the probability of detecting a 1 on a qubit initially in state $|0\rangle$ is 5%, and the probability of observing a 0 on a qubit in state $|1\rangle$ is 23%. Given knowledge of A , one can compute an error-mitigated empirical mean value of an observable O as

$$\xi = M^{-1} \sum_{i=1}^M \sum_x O(x) \langle x|A^{-1}|s^i\rangle, \quad (\text{A3})$$

where M represents the number of independent experimental shots, and $s^i \in \{0, 1\}^n$ is the sequence of measurement outcomes from the i -th experiment. The reconstruction of A (and thereby A^{-1}) through sample circuit measurements lies at the heart of REM. While determining the complete noise matrix A typically requires calibration for each potential input state, as suggested

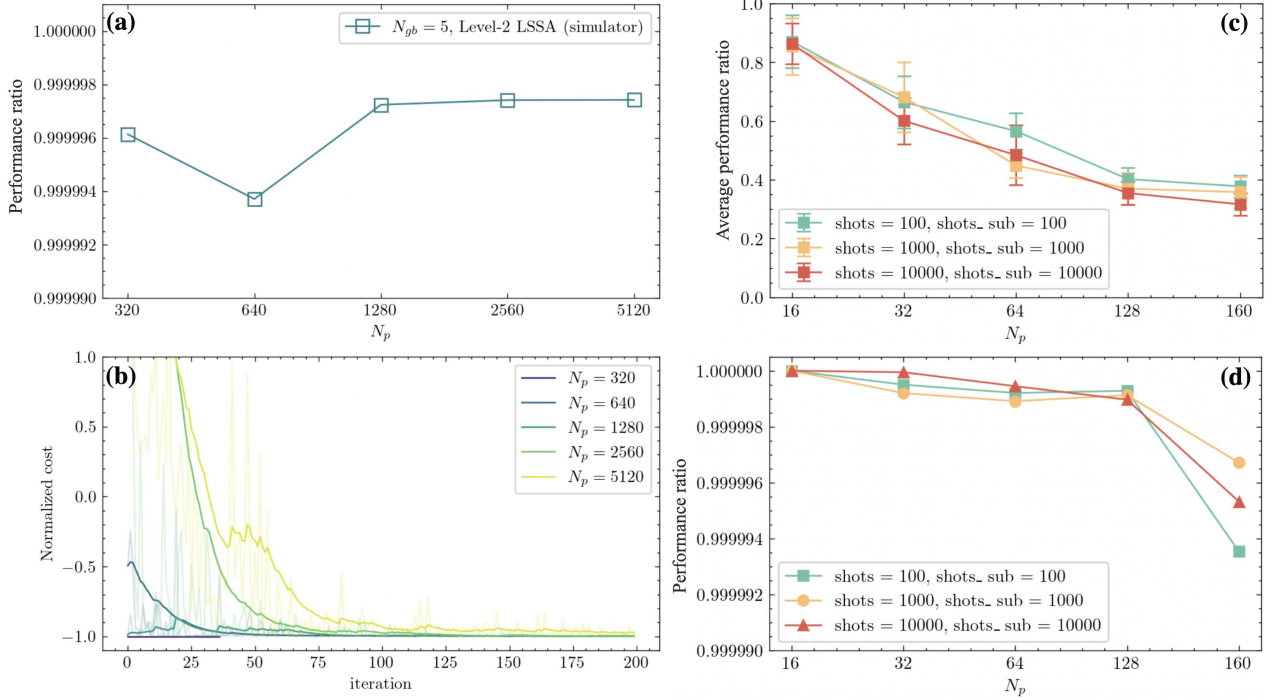


FIG. 14. (a) Performance ratio of the portfolio optimization problems up to $N_p = 5120$ solved by `ibmq_qasm-simulator` using 5 qubits. For each data point, the best result among its 3 attempts is picked. (b) Normalized cost during the optimization process for different values of N_p from $N_p = 320$ to $N_p = 5120$. The data are smoothed to better observe the behavior, while the original data are faded in the background. (c) Average performance ratios of the random Ising problems for different measurement shots by `ibmq_qasm-simulator`, with $N_g = 5$, $N_s = \lceil N_p/5 \rceil$ and $N_p \in \{16, 32, 64, 128, 160\}$. 100 problems are tested for each data point. (d) Performance ratios of the portfolio optimization problem for different measurement shots by `ibmq_qasm-simulator`. Parameters N_g , N_p and N_s are the same as those in (c), but for each data point, the best result among its 20 attempts is picked.

in [45], it's also feasible to estimate A by sampling subsets of the full binary state space using continuous time markov processes (CTMP), providing a nuanced alternative method.

Fig. 12 contrasts the unmitigated and mitigated performance ratios for fully-connected random Ising problems obtained from the 7-qubit `ibmq_lagos`. The manageable scale of a 7-qubit system allows for a full calibration of each possible input state to reconstruct A within a reasonable timeframe, which is the approach we adopt. With $N_g = 7$, $N_p \in \{16, 32, 64\}$, and $N_s = \lceil 2N_p/N_g \rceil$. For the subsystem problems, a single layer QAOA is used with 5 iterations of the Nelder-Mead optimization, and 8192 measurement shots are used for each calculation of the expectation value. The mitigated results show a visible improvement compared to the unmitigated results. For the portfolio optimization problems satisfying the good performance condition in Sec. IIIB, LSSA already yields very good results by quantum hardware as shown in Fig. 5(b). So the effect of noise is likely too minor to affect the results, and thus, no further error mitigation is applied to this case.

Appendix B: Effect of different circuit structures of VQE

The VQE circuit structure (ansatz) used in the subsystem coefficient optimization stage is also one of the hyperparameters that may affect our result. The structure of the quantum circuit of our VQE consists of alternating entanglement layers and rotation layers as shown in Fig. 1 and Fig. 13(a). The p -layer structure in Fig. 13 means that the circuit in the dashed box is repeated p times, and labels “fe” and “le” indicate that the entanglement strategies of the entanglement layers in the dashed box being full entanglement [the upper graph in Fig. 13(a)] and linear entanglement [the lower graph in Fig. 13(a)], respectively. In the paper, our main result is computed by the “2-layer fe” structure described in Fig. 1 and Fig. 13(a). In this appendix, we tune the type of the entanglement strategies and the number of the parameterized circuit layers in the dashed box for solving the problems with $N_p = 20$, $N_g = 5$ and $N_s = 8$ by exact diagonalization for subsystems and `ibmq_qasm-simulator` for coefficient optimization. For the random Ising problem, 100 problems are solved and averaged, while for the portfolio optimization problem, we pick the best result

of their 10 attempts for the same problem. As shown in Fig. 13(b), the average performance ratios for the cases with linear entanglement are slightly better than those with full entanglement in simulations on simulators. Here the performance ratio is defined as $R_{\text{ar}} = \frac{\text{LSSA GSE}}{\text{Exact GSE}}$. For the problem we consider, increasing the number of parameterized circuit layers help slightly the performance in the average performance ratios for both the cases with full entanglement and linear entanglement, but the effect is less obvious for the cases with linear entanglement. Both the quantum circuits with linear entanglement and full entanglement layers have comparable performance. But if only one rotation layer and no entanglement layer is used for the circuit, then the performance in the average performance ratios degrades.

Appendix C: Level-2 approximation

Dividing the full problem into subsystems makes it possible to solve for the problem with size larger than that of our quantum hardware. Moreover, it may be possible to decompose the subsystems into sub-subsystems, and this will significantly increase the maximal problem size we can solve with our available quantum hardware. Let us call the procedure that use LSSA once the level-1 approximation since the problem is decomposed once and the subsystems are in the same size. In the level-1 approximation, an N_{gb} -qubit gate-based can solve QUBO problems up to $N_{\text{gb}}2^{N_{\text{gb}}}$ variables. If we further consider $N_{\text{gb}}2^{N_{\text{gb}}}$ as our subsystem size, with $2^{N_{\text{gb}}}$ subsystems that can be encoded into VQE for coefficient optimization, we can solve QUBO problems with sizes $N_p = N_{\text{gb}}2^{2N_{\text{gb}}}$ in total. We call this further decomposition the level-2 approximation. This means that with only 5-qubit gate-based quantum computers, the maximal size of the problems that can be solve in the level-2 approximation is $N_p = 5 \times 2^{10} = 5120$. Again, this means that we will have $2^{2N_{\text{gb}}}$ subsystems to be solved before dealing with the full problem Hamiltonian for the VQE coefficient optimization. This could be a quite large amount and could make LSSA too slow to be useful. However, this may be conquered by solving the subsystem problems on several quantum computers in parallel, due to the fact that there is no causality between the subsystems in the same level. In practical terms, it would be much easier to have hundreds or dozens of very reliable 5-qubit or 10-qubit quantum computers than to have or develop a single reliable quantum computer with several thousands qubits. Quantum computer clusters utilizing the concept of LSSA could be one of the possible scenarios that the real-world use case of

quantum computing applications could be found in the future. To demonstrate what level-2 LSSA is capable of, in this appendix, we use it to solve, as an example, the portfolio optimization problems with stock data sizes $N_p \in \{320, 640, 1280, 2560, 5120\}$. With 5-qubit quantum computers available, the subsystem sizes $N_{\text{gb}}^{(1)} = 160$ and $N_{\text{gb}}^{(2)} = 5$ in the level-1 and level-2 approximations mean that the first level of LSSA divides the problem into subsystem problems with sizes $N_{\text{gb}}^{(1)} = 160$, then the second level of LSSA further divides the problem from subsystem size $N_{\text{gb}}^{(1)} = 160$ to $N_{\text{gb}}^{(2)} = 5$, with $N_s^{(1)} = \lceil N_p / N_{\text{gb}}^{(1)} \rceil$ and $N_s^{(2)} = 2^{N_{\text{gb}}^{(2)}} = 32$. In Fig. 14, we show the performance ratio and normalized cost during the optimization process calculated by simulators for problem sizes mentioned above. The performance ratio here is also defined as $R_{\text{ar}} = \frac{\text{LSSA GSE}}{\text{Dwave-tabu GSE}}$. It is interesting to observe that the performance ratio still holds up well in the portfolio optimization problems even if we use level-2 LSSA.

Appendix D: Device information

In our demonstrations on real hardware, we have used IBM Quantum 27-qubit `ibm_cairo`, 27-qubit `ibm_auckland`, and 16-qubit `ibmq_guadalupe` gate-based quantum systems, and 5627-qubit D-Wave Advantage 4 quantum annealer machine. The layouts of the IBM Quantum processors we used are shown in Fig. 15. Since the characteristic data (e.g. qubit relaxation time and gate error) of a quantum computer could change over time, here we list the data at the time when the demonstrations were performed. Tables I, II, and III list the qubit relaxation time T_1 , dephasing time T_2 , qubit frequency, single-qubit gate error and readout error data of `ibm_cairo`, `ibm_auckland`, and `ibmq_guadalupe`, respectively. Tables IV, V, and VI list the two-qubit coupling and corresponding CNOT gate error and gate length data of `ibm_cairo`, `ibm_auckland`, and `ibmq_guadalupe`, respectively. In the case of D-Wave device, the topology represents the pattern of physical connections for qubits and the couplers between them. The type of the layout topology of D-Wave Advantage 4 is called Pegasus, which has increased connectivity between the qubits of quantum annealing hardware compared to the graph Chimera. The detailed definitions of Pegasus and Chimera can be founded in [48]. We first show a relatively simple P5 Pegasus graph in Fig. 16(a), and then show the P16 Pegasus layout of the D-Wave Advantage 4 in Fig. 16(b), which has degree (the number of couplers per qubit) 15 and contains 5627 qubits and 40679 couplers [47].

[1] IBM Quantum, <https://quantum-computing.ibm.com> (n.d.).

[2] Amazon Braket, <https://aws.amazon.com/braket> (2022).

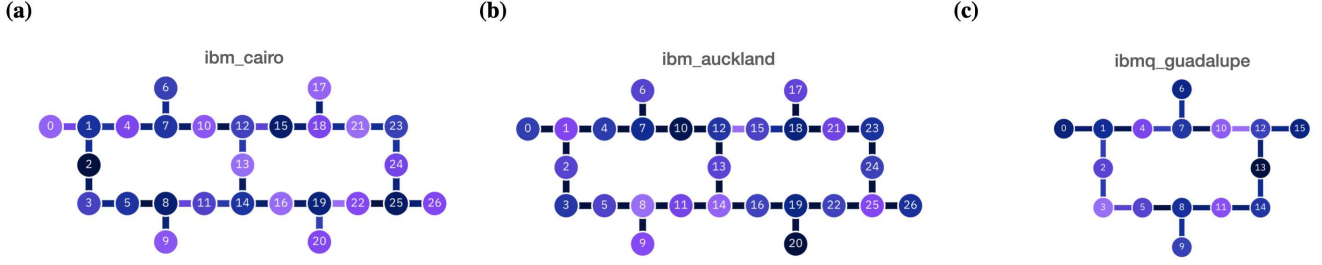


FIG. 15. Device layouts of the IBM quantum processors: (a) The 27-qubit system `ibm_cairo`, (b) The 27-qubit system `ibmq_auckland`, and (c) The 16-qubit system `ibmq_guadalupe`. From IBM Quantum service.

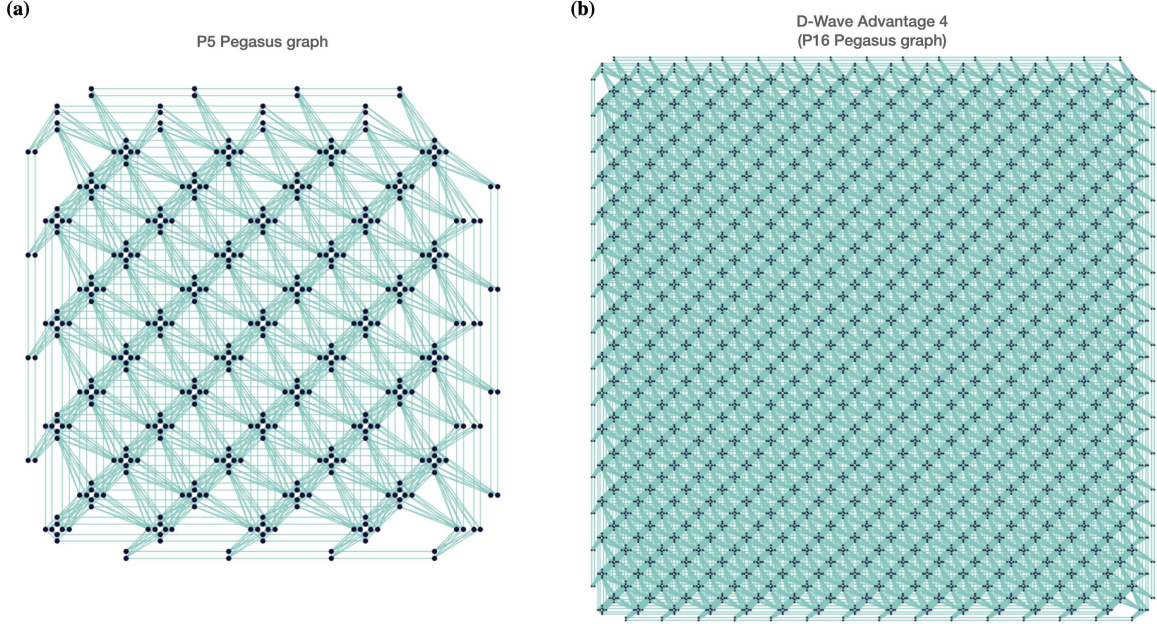


FIG. 16. Layout graphs of D-Wave devices: (a) A P5 Pegasus graph containing 448 qubits and 2864 couplers. (b) Topology of D-Wave Advantage 4, a P16 Pegasus graph with degree (the number of couplers per qubit) 15, containing 5627 qubits and 40679 couplers. From [47].

- [3] D-Wave, <https://www.dwavesys.com> (2022).
- [4] Z. Bian, F. Chudak, W. Macready, A. Roy, R. Sebastiani, and S. Varotti, [Solving sat and maxsat with a quantum annealer: Foundations, encodings, and preliminary results](#) (2018), [arXiv:1811.02524](#).
- [5] A. Lucas, Ising formulations of many np problems, *Frontiers in Physics* **2**, 10.3389/fphy.2014.00005 (2014).
- [6] J. Cohen, A. Khan, and C. Alexander, [Portfolio optimization of 40 stocks using the dwave quantum annealer](#) (2020), [arXiv:2007.01430](#).
- [7] S. Mugel, C. Kuchkovsky, E. Sánchez, S. Fernández-Lorenzo, J. Luis-Hita, E. Lizaso, and R. Orús, Dynamic portfolio optimization with real datasets using quantum processors and quantum-inspired tensor networks, *Phys. Rev. Research* **4**, 013006 (2022).
- [8] N. N. Hegade, P. Chandarana, K. Paul, X. Chen, F. Albarrán-Arriagada, and E. Solano, [Portfolio optimization with digitized-counterdiabatic quantum algorithms](#) (2021), [arXiv:2112.08347](#).
- [9] S. Palmer, S. Sahin, R. Hernandez, S. Mugel, and R. Orus, [Quantum portfolio optimization with investment bands and target volatility](#) (2021), [arXiv:2106.06735](#).
- [10] S. J. Weinberg, F. Sanches, T. Ide, K. Kamiya, and R. Correll, [Supply chain logistics with quantum and classical annealing algorithms](#) (2022), [arXiv:2205.04435](#).
- [11] C. Dalyac, L. Henriet, E. Jeandel, W. Lechner, S. Perdrix, M. Porcheron, and M. Veshchezerova, Qualifying quantum approaches for hard industrial optimization problems. a case study in the field of smart-charging of electric vehicles, *EPJ Quantum Technology* **8**, 12 (2021).
- [12] A. E. Moylett, N. Linden, and A. Montanaro, Quantum speedup of the traveling-salesman problem for bounded-degree graphs, *Phys. Rev. A* **95**, 032323 (2017).
- [13] Ö. Salehi, A. Glos, and J. A. Mischczak, Unconstrained binary models of the travelling salesman problem vari-

Qubit	T1 (μ s)	T2 (μ s)	Freq. (GHz)	Gate error	Readout error
0	105.356	64.329	5.2574	0.017%	1.13%
1	23.375	13.391	5.0358	0.044%	1.64%
2	78.662	66.801	4.9109	0.019%	1.05%
3	63.562	105.984	5.1185	0.026%	0.93%
4	65.297	27.253	5.1856	0.020%	1.29%
5	118.752	63.980	5.0461	0.037%	0.73%
6	151.008	145.692	5.0989	0.037%	1.00%
7	85.363	110.666	5.0481	0.047%	1.67%
8	99.307	103.506	4.9689	0.016%	1.20%
9	114.280	81.611	5.2275	0.016%	3.15%
10	85.609	30.319	5.2335	0.014%	0.62%
11	83.719	124.963	5.1327	0.026%	0.67%
12	150.588	265.045	5.1149	0.018%	0.77%
13	94.137	159.659	5.2820	0.020%	0.73%
14	103.875	236.430	5.0437	0.020%	0.74%
15	126.853	188.488	4.9626	0.032%	0.99%
16	89.114	146.802	5.2769	0.026%	2.96%
17	127.372	70.219	5.2622	0.048%	1.23%
18	99.137	6.412	5.1860	0.091%	1.98%
19	154.641	171.512	4.9781	0.037%	5.33%
20	139.907	16.904	5.1971	0.125%	1.06%
21	114.308	104.903	5.2843	0.020%	0.48%
22	125.717	134.834	5.2002	0.019%	0.96%
23	169.922	196.208	5.1024	0.035%	1.01%
24	148.204	204.221	5.1768	0.046%	1.15%
25	155.145	185.784	4.9538	0.010%	0.64%
26	129.085	55.617	5.1920	0.015%	1.63%

TABLE I. Qubit relaxation time T_1 , dephasing time T_2 , qubit frequency, single-qubit gate errors, and readout error data of the 27-qubit machine `ibm_cairo` downloaded from the IBM Quantum system service at the time when the demonstration was performed.

- ants for quantum optimization, [Quantum Information Processing](#) **21**, 67 (2022).
- [14] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O’Brien, A variational eigenvalue solver on a photonic quantum processor, [Nature Communications](#) **5**, 4213 (2014).
- [15] X. Liu, A. Angone, R. Shaydulin, I. Safro, Y. Alexeev, and L. Cincio, Layer vqe: A variational approach for combinatorial optimization on noisy quantum computers, [IEEE Transactions on Quantum Engineering](#) **3**, 1–20 (2022).
- [16] D. Amaro, C. Modica, M. Rosenkranz, M. Fiorentini, M. Benedetti, and M. Lubasch, Filtering variational quantum algorithms for combinatorial optimization, [Quantum Science and Technology](#) **7**, 015021 (2022).
- [17] E. Farhi, J. Goldstone, and S. Gutmann, [A quantum approximate optimization algorithm](#) (2014), [arXiv:1411.4028](#).
- [18] A. Galda, X. Liu, D. Lykov, Y. Alexeev, and I. Safro, Transferability of optimal qaoa parameters between random graphs (2021), [arXiv:2106.07531 \[quant-ph\]](#).
- [19] Z. He, R. Shaydulin, S. Chakrabarti, D. Herman, C. Li, Y. Sun, and M. Pistoia, Alignment between initial state and mixer improves qaoa performance for constrained optimization, [npj Quantum Information](#) **9**, 10.1038/s41534-023-00787-5 (2023).
- [20] J. Golden, A. Bärttschi, D. O’Malley, and S. Eidenbenz, Numerical evidence for exponential speed-up of qaoa over unstructured search for approximate constrained optimization, in [2023 IEEE International Conference on Quantum Computing and Engineering \(QCE\)](#) (IEEE, 2023).
- [21] J. Golden, A. Bärttschi, D. O’Malley, and S. Eidenbenz, The quantum alternating operator ansatz for satisfiability problems, in [2023 IEEE International Conference on Quantum Computing and Engineering \(QCE\)](#) (IEEE, 2023).
- [22] R. Shaydulin, C. Li, S. Chakrabarti, M. DeCross, D. Herman, N. Kumar, J. Larson, D. Lykov, P. Minssen, Y. Sun, Y. Alexeev, J. M. Dreiling, J. P. Gaebler, T. M. Gatterman, J. A. Gerber, K. Gilmore, D. Gresh, N. Hewitt, C. V. Horst, S. Hu, J. Johansen, M. Matheny, T. Mengle, M. Mills, S. A. Moses, B. Neyenhuis, P. Siegfried, R. Yalovetzky, and M. Pistoia, Evidence of scaling advantage for the quantum approximate optimization algorithm on a classically intractable problem (2023), [arXiv:2308.02342 \[quant-ph\]](#).
- [23] E. Pelofske, A. Bärttschi, J. Golden, and S. Eidenbenz, High-round qaoa for max k -sat on trapped ion niscq devices, in [2023 IEEE International Conference on Quantum Computing and Engineering \(QCE\)](#) (IEEE, 2023).
- [24] E. Pelofske, A. Bärttschi, L. Cincio, J. Golden, and S. Eidenbenz, Scaling whole-chip qaoa for higher-order ising spin glass models on heavy-hex graphs (2023), [arXiv:2312.00997 \[quant-ph\]](#).
- [25] R. Shaydulin and M. Pistoia, Qaoa with $n \cdot p \geq 200$ (2023), [arXiv:2303.02064 \[quant-ph\]](#).

Qubit	T1 (μ s)	T2 (μ s)	Freq. (GHz)	Gate error	Readout error
0	173.790	68.007	4.9330	0.021%	0.65%
1	174.824	215.223	5.0738	0.030%	0.88%
2	97.702	183.570	5.0059	0.020%	0.87%
3	222.540	63.538	4.8972	0.024%	1.50%
4	56.023	223.711	4.9203	0.017%	0.82%
5	138.836	78.402	4.9928	0.021%	0.89%
6	92.703	137.724	5.0065	0.044%	1.24%
7	168.171	187.972	4.8284	0.016%	0.74%
8	153.591	96.525	5.2036	0.020%	1.00%
9	164.735	61.682	5.0885	0.020%	0.66%
10	97.861	45.922	4.7269	0.024%	0.57%
11	179.578	140.51	5.0552	0.027%	0.56%
12	10.769	12.379	4.8896	0.153%	1.89%
13	134.829	18.540	5.0168	0.028%	1.14%
14	115.201	134.717	5.1671	0.024%	0.85%
15	133.747	161.793	4.9891	0.179%	1.03%
16	189.344	190.667	4.9695	0.025%	0.67%
17	126.372	117.635	5.0254	0.026%	1.27%
18	194.773	254.374	4.7879	0.024%	0.96%
19	205.831	24.949	4.8063	0.020%	0.97%
20	90.326	140.011	4.6919	0.019%	10.01%
21	143.962	120.571	5.0463	0.019%	0.87%
22	175.778	77.591	4.9684	0.028%	0.79%
23	85.360	46.600	4.8679	0.023%	1.12%
24	130.000	49.416	4.9560	0.041%	3.14%
25	161.979	257.114	5.0768	0.015%	0.85%
26	98.704	49.203	4.8559	0.024%	0.64%

TABLE II. Qubit relaxation time T_1 , dephasing time T_2 , qubit frequency, single-qubit gate errors, and readout error data of the 27-qubit machine `ibm_auckland` downloaded from the IBM Quantum system service at the time when the demonstration was performed.

Qubit	T1 (μ s)	T2 (μ s)	Freq. (GHz)	Gate error	Readout error
0	78.859	19.423	5.1135	0.115%	3.09%
1	117.329	140.678	5.1608	0.046%	2.51%
2	86.476	101.725	5.3105	0.032%	1.69%
3	69.082	79.001	5.4605	0.296%	22.74%
4	97.591	117.840	5.3534	0.034%	2.42%
5	109.816	121.331	5.3033	0.029%	1.68%
6	93.002	14.737	5.1364	0.044%	1.77%
7	82.319	174.260	5.2028	0.035%	1.54%
8	96.776	81.175	5.1745	0.028%	3.08%
9	101.589	60.979	5.2537	0.065%	6.50%
10	75.128	146.406	5.4268	0.027%	1.28%
11	124.369	97.742	5.3898	0.049%	6.40%
12	104.287	80.710	5.2637	0.034%	1.96%
13	96.336	85.164	5.0380	0.044%	1.62%
14	118.148	136.963	5.2050	0.026%	1.75%
15	65.321	162.541	5.1264	0.021%	1.53%

TABLE III. Qubit relaxation time T_1 , dephasing time T_2 , qubit frequency, single-qubit gate errors, and readout error data of the 16-qubit machine `ibmq_guadalupe` downloaded from the IBM Quantum system service at the time when the demonstration was performed.

- [26] M. P. Harrigan, K. J. Sung, M. Neeley, K. J. Satzinger, F. Arute, K. Arya, J. Atalaya, J. C. Bardin, R. Barends, S. Boixo, M. Broughton, B. B. Buckley, D. A. Buell, B. Burkett, N. Bushnell, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, S. Demura, A. Dunsworth, D. Eppens, A. Fowler, B. Foxen, C. Gidney, M. Giustina, R. Graff, S. Habegger, A. Ho, S. Hong, T. Huang, L. B. Ioffe, S. V. Isakov, E. Jeffrey, Z. Jiang, C. Jones, D. Kafri, K. Kechedzhi, J. Kelly, S. Kim, P. V. Klimov, A. N. Korotkov, F. Kostritsa, D. Landhuis, P. Laptev,

Coupling pair	Gate error	Gate length(ns)
[0, 1]	2.119%	554.67
[1, 0]	2.119%	533.33
[1, 2]	0.912%	320.00
[1, 4]	3.590%	362.67
[2, 1]	0.912%	341.33
[2, 3]	0.998%	220.44
[3, 2]	0.998%	199.11
[3, 5]	0.546%	277.33
[4, 1]	3.590%	384.00
[4, 7]	0.796%	263.11
[5, 3]	0.546%	320.00
[5, 8]	1.094%	174.22
[6, 7]	1.254%	305.78
[7, 4]	0.796%	284.44
[7, 6]	1.254%	327.11
[7, 10]	100%	234.67
[8, 5]	1.094%	216.89
[8, 9]	0.616%	256.00
[8, 11]	4.714%	405.33
[9, 8]	0.616%	234.67
[10, 7]	100%	213.33
[10, 12]	0.545%	224.00
[11, 8]	4.714%	426.67
[11, 14]	1.105%	188.44
[12, 10]	0.545%	266.67
[12, 13]	1.376%	391.11
[12, 15]	0.957%	512.00
[13, 12]	1.376%	412.44
[13, 14]	0.590%	220.44
[14, 11]	1.105%	231.11
[14, 13]	0.590%	263.11
[14, 16]	0.490%	245.33
[15, 12]	0.957%	490.67
[15, 18]	1.251%	277.33
[16, 14]	0.490%	202.67
[16, 19]	0.777%	270.22
[17, 18]	3.945%	248.89
[18, 15]	1.251%	234.67
[18, 17]	3.945%	270.22
[18, 21]	4.957%	316.44
[19, 16]	0.777%	312.89
[19, 20]	0.950%	426.67
[19, 22]	0.531%	270.22
[20, 19]	0.950%	405.33
[21, 18]	4.957%	273.78
[21, 23]	0.848%	316.44
[22, 19]	0.531%	227.56
[22, 25]	0.663%	184.89
[23, 21]	0.848%	359.11
[23, 24]	0.953%	209.78
[24, 23]	0.953%	167.11
[24, 25]	0.526%	263.11
[25, 22]	0.663%	227.56
[25, 24]	0.526%	305.78
[25, 26]	0.602%	248.89
[26, 25]	0.602%	206.22

TABLE IV. Two-qubit CNOT gate error and gate length data of the 27-qubit machine `ibm_cairo` downloaded from the IBM Quantum system service at the time when the demonstration was performed.

Coupling pair	Gate error	Gate length(ns)
[0, 1]	1.867%	1528.89
[1, 0]	1.867%	1564.44
[1, 2]	0.677%	170.67
[1, 4]	0.854%	497.78
[2, 1]	0.677%	241.78
[2, 3]	0.620%	224.00
[3, 2]	0.620%	295.11
[3, 5]	0.601%	277.33
[4, 1]	0.854%	462.22
[4, 7]	0.352%	195.56
[5, 3]	0.601%	206.22
[5, 8]	0.394%	320.00
[6, 7]	1.701%	640.00
[7, 4]	0.352%	266.67
[7, 6]	1.701%	675.56
[7, 10]	0.832%	280.89
[8, 5]	0.394%	248.89
[8, 9]	0.648%	504.89
[8, 11]	0.692%	369.78
[9, 8]	0.648%	469.33
[10, 7]	0.832%	352.00
[10, 12]	4.888%	704.00
[11, 8]	0.692%	405.33
[11, 14]	0.655%	369.78
[12, 10]	4.888%	739.56
[12, 13]	2.483%	384.00
[12, 15]	5.996%	590.22
[13, 12]	2.483%	348.44
[13, 14]	1.661%	430.22
[14, 11]	0.655%	334.22
[14, 13]	1.661%	359.11
[14, 16]	0.663%	355.56
[15, 12]	5.996%	554.67
[15, 18]	1.309%	654.22
[16, 14]	0.663%	391.11
[16, 19]	1.018%	711.11
[17, 18]	0.772%	440.89
[18, 15]	1.309%	618.67
[18, 17]	0.772%	476.44
[18, 21]	0.708%	391.11
[19, 16]	1.018%	746.67
[19, 20]	0.652%	398.22
[19, 22]	1.697%	590.22
[20, 19]	0.652%	433.78
[21, 18]	0.708%	355.56
[21, 23]	1.039%	540.44
[22, 19]	1.697%	554.67
[22, 25]	1.366%	426.67
[23, 21]	1.039%	576.00
[23, 24]	0.941%	348.44
[24, 23]	0.941%	384.00
[24, 25]	0.778%	440.89
[25, 22]	1.366%	462.22
[25, 24]	0.778%	405.33
[25, 26]	0.502%	348.44
[26, 25]	0.502%	384.00

TABLE V. Two-qubit CNOT gate error and gate length data of the 27-qubit machine `ibm_auckland` downloaded from the IBM Quantum system service at the time when the demonstration was performed.

Coupling pair	Gate error	Gate length(ns)
[0, 1]	1.587%	334.22
[1, 0]	1.587%	369.78
[1, 2]	1.068%	504.89
[1, 4]	1.999%	298.67
[2, 1]	1.068%	469.33
[2, 3]	2.781%	483.56
[3, 2]	2.781%	519.11
[3, 5]	2.174%	583.11
[4, 1]	1.999%	263.11
[4, 7]	1.251%	512.00
[5, 3]	2.174%	547.56
[5, 8]	0.747%	270.22
[6, 7]	0.890%	405.33
[7, 4]	1.251%	476.44
[7, 6]	0.890%	369.78
[7, 10]	0.669%	298.67
[8, 5]	0.747%	305.78
[8, 9]	1.877%	412.44
[8, 11]	1.468%	348.44
[9, 8]	1.877%	448.00
[10, 7]	0.669%	263.11
[10, 12]	1.635%	775.11
[11, 8]	1.468%	312.89
[11, 14]	1.364%	348.44
[12, 10]	1.635%	739.56
[12, 13]	0.804%	348.44
[12, 15]	0.629%	376.89
[13, 12]	0.804%	384.00
[13, 14]	0.879%	398.22
[14, 11]	1.364%	384.00
[14, 13]	0.879%	362.67
[15, 12]	0.629%	412.44

TABLE VI. Two-qubit CNOT gate error and gate length data of the 16-qubit machine `ibmq_guadalupe` downloaded from the IBM Quantum system service at the time when the demonstration was performed.

- M. Lindmark, M. Leib, O. Martin, J. M. Martinis, J. R. McClean, M. McEwen, A. Megrant, X. Mi, M. Mohseni, W. Mruczkiewicz, J. Mutus, O. Naaman, C. Neill, F. Neukart, M. Y. Niu, T. E. O'Brien, B. O'Gorman, E. Ostby, A. Petukhov, H. Putterman, C. Quintana, P. Roushan, N. C. Rubin, D. Sank, A. Skolik, V. Smelyanskiy, D. Strain, M. Streif, M. Szalay, A. Vainsencher, T. White, Z. J. Yao, P. Yeh, A. Zalcman, L. Zhou, H. Neven, D. Bacon, E. Lucero, E. Farhi, and R. Babbush, Quantum approximate optimization of non-planar graph problems on a planar superconducting processor, *Nature Physics* **17**, 332–336 (2021).
- [27] E. Farhi, J. Goldstone, S. Gutmann, and L. Zhou, The quantum approximate optimization algorithm and the sherrington-kirkpatrick model at infinite size, *Quantum* **6**, 759 (2022).
- [28] S. Hadfield, Z. Wang, B. O'Gorman, E. Rieffel, D. Venturelli, and R. Biswas, From the quantum approximate optimization algorithm to a quantum alternating operator ansatz, *Algorithms* **12**, 34 (2019).
- [29] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm (2014), [arXiv:1411.4028 \[quant-ph\]](https://arxiv.org/abs/1411.4028).
- [30] J. Li, M. Alam, and S. Ghosh, Large-scale quantum approximate optimization via divide-and-conquer (2021), [arXiv:2102.13288](https://arxiv.org/abs/2102.13288).
- [31] Z. Zhou, Y. Du, X. Tian, and D. Tao, Qaoa-in-qaoa: solving large-scale maxcut problems on small quantum machines (2022), [arXiv:2205.11762](https://arxiv.org/abs/2205.11762).
- [32] Y. Zhang, L. Cincio, C. F. A. Negre, P. Czarnik, P. Coles, P. M. Anisimov, S. M. Mniszewski, S. Tretiak, and P. A. Dub, Variational quantum eigensolver with reduced circuit complexity (2021), [arXiv:2106.07619](https://arxiv.org/abs/2106.07619).
- [33] B. Tan, M.-A. Lemonde, S. Thanasilp, J. Tangpanitanon, and D. G. Angelakis, Qubit-efficient encoding schemes for binary optimisation problems, *Quantum* **5**, 454 (2021).
- [34] A. Suau, J. Nelson, M. Vuffray, A. Y. Lokhov, L. Cincio, and C. Coffrin, Single-qubit cross platform comparison of quantum computing hardware (2021), [arXiv:2108.11334 \[quant-ph\]](https://arxiv.org/abs/2108.11334).
- [35] E. Pelofske, J. Golden, A. Bartschi, D. O'Malley, and S. Eidenbenz, Sampling on nisq devices: “who’s the fairest one of all?”, in *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)* (IEEE, 2021).
- [36] E. Pelofske, A. Bartschi, and S. Eidenbenz, Quantum annealing vs. qaoa: 127 qubit higher-order ising problems on nisq computers, in *High Performance Computing* (Springer Nature Switzerland, 2023) p. 240–258.
- [37] M. Booth and S. P. Reinhardt, Partitioning optimization problems for hybrid classical/quantum execution, D-Wave Technical Report Series **14-1006A-A** (2017).
- [38] F. Phillipson and H. S. Bhatia, Portfolio optimisation using the d-wave quantum annealer (2020), [arXiv:2012.01121](https://arxiv.org/abs/2012.01121).
- [39] J. Raymond, R. Stevanovic, W. Bernoudy, K. Boothby, C. C. McGeoch, A. J. Berkley, P. Farré, J. Pasvolsky, and A. D. King, Hybrid quantum annealing for larger-than-qpu lattice-structured problems, *ACM Transactions on Quantum Computing* **4**, 1–30 (2023).
- [40] R. Shaydulin, H. Ushijima-Mwesigwa, I. Safro, S. Mniszewski, and Y. Alexeev, Network community detection on small quantum computers, *Advanced Quantum Technologies* **2**, 1900029 (2019).
- [41] H. Markowitz, Portfolio selection, *The Journal of Finance* **7**, 77 (1952).
- [42] D-Wave, The advantage system: Performance update, https://www.dwavesys.com/media/kjtlcemb/14-1054a-a_advantage_system_performance_update.pdf (2021).
- [43] Yahoo Finance, <https://finance.yahoo.com/> (2022).
- [44] E. van den Berg, Z. K. Mineev, and K. Temme, Model-free readout-error mitigation for quantum expectation values, *Phys. Rev. A* **105**, 032620 (2022).
- [45] S. Bravyi, S. Sheldon, A. Kandala, D. C. McKay, and J. M. Gambetta, Mitigating measurement errors in multiqubit experiments, *Phys. Rev. A* **103**, 042605 (2021).
- [46] K. Temme, S. Bravyi, and J. M. Gambetta, Error mitigation for short-depth quantum circuits, *Phys. Rev. Lett.* **119**, 180509 (2017).
- [47] D-Wave, D-wave qpu-specific characteristics, https://docs.dwavesys.com/docs/latest/doc_physical_properties.html#doc-qpu-characteristics (n.d.).
- [48] N. Dattani, S. Szalay, and N. Chancellor, Pegasus: The second connectivity graph for large-scale quantum annealing hardware (2019), [arXiv:1901.07636](https://arxiv.org/abs/1901.07636).